PCT

WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : G06F

A2

(11) International Publication Number: WO 99/41653

(43) International Publication Date: 19 August 1999 (19.08.99)

(21) International Application Number: PCT/US99/02676

(22) International Filing Date: 8 February 1999 (08.02.99)

(30) Priority Data:
09/025,321 12 February 1998 (12.02.98) US

(71) Applicant (*for all designated States except US*): IAMETER, INCORPORATED [US/US]; Suite 565, 901 Mariner's Island Boulevard, San Mateo, CA 94404 (US).

(72) Inventor; and
(75) Inventor/Applicant (*for US only*): WHITING-O'KEEFE, Quinn [US/US]; 2202 Edgewood Road, Redwood City, CA 94062 (US).

(74) Agents: PARSONS, Gerald, P. et al.; Majestic, Parsons, Siebert & Hsue P.C., Suite 1100, Four Embarcadero Center, San Francisco, CA 94111-4106 (US).

(81) Designated States: AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CU, CZ, DE, DK, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, UA, UG, US, UZ, VN, YU, ZW, ARIPO patent (GH, GM, KE, LS, MW, SD, SZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published
Without international search report and to be republished upon receipt of that report.

(54) Title: TECHNIQUES FOR ESTIMATING CHARGES OF DELIVERING HEALTHCARE SERVICES THAT TAKE COMPLICATING FACTORS INTO ACCOUNT

(57) Abstract

The expected charges for treating a patient are estimated by the use of linear regression techniques wherein variables and coefficients of estimate models are built from historic patient data of all episode types (in-patient, office visits, etc.) and which include secondary and collateral illnesses that greatly affect the cost of treating a patient for the primary illness. In addition to evaluating the primary illness, the evaluation of collateral illness is also made possible since the processing techniques are not simply averaging data for the primary illness as has been done in the past. This allows physicians to understand the magnitude and types of resource usage that are appropriate for particular illnesses. The present invention allows better management of many aspects of the healthcare system by significantly reducing a component of the difference between the estimate and actual charges for treating a particular illness of a patient that is due to the general state of health of the patient. This then allows the remaining difference to be attributed to differences in the way that health service providers treat each illness of a patient. Estimates are made for the expected cost of providing specific episodes of care, to treat each illness in its entirety by use of both in-patient and out-patient services, and to provide the specific procedures used for the appropriate management of each illness of each patient.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	ES	Spain	LS	Lesotho	SI	Slovenia
AM	Armenia	FI	Finland	LT	Lithuania	SK	Slovakia
AT	Austria	FR	France	LU	Luxembourg	SN	Senegal
AU	Australia	GA	Gabon	LV	Latvia	SZ	Swaziland
AZ	Azerbaijan	GB	United Kingdom	MC	Monaco	TD	Chad
BA	Bosnia and Herzegovina	GE	Georgia	MD	Republic of Moldova	TG	Togo
BB	Barbados	GH	Ghana	MG	Madagascar	TJ	Tajikistan
BE	Belgium	GN	Guinea	MK	The former Yugoslav Republic of Macedonia	TM	Turkmenistan
BF	Burkina Faso	GR	Greece			TR	Turkey
BG	Bulgaria	HU	Hungary	ML	Mali	TT	Trinidad and Tobago
BJ	Benin	IE	Ireland	MN	Mongolia	UA	Ukraine
BR	Brazil	IL	Israel	MR	Mauritania	UG	Uganda
BY	Belarus	IS	Iceland	MW	Malawi	US	United States of America
CA	Canada	IT	Italy	MX	Mexico	UZ	Uzbekistan
CF	Central African Republic	JP	Japan	NE	Niger	VN	Viet Nam
CG	Congo	KE	Kenya	NL	Netherlands	YU	Yugoslavia
CH	Switzerland	KG	Kyrgyzstan	NO	Norway	ZW	Zimbabwe
CI	Côte d'Ivoire	KP	Democratic People's Republic of Korea	NZ	New Zealand		
CM	Cameroon			PL	Poland		
CN	China	KR	Republic of Korea	PT	Portugal		
CU	Cuba	KZ	Kazakstan	RO	Romania		
CZ	Czech Republic	LC	Saint Lucia	RU	Russian Federation		
DE	Germany	LI	Liechtenstein	SD	Sudan		
DK	Denmark	LK	Sri Lanka	SE	Sweden		
EE	Estonia	LR	Liberia	SG	Singapore		

TECHNIQUES FOR ESTIMATING CHARGES OF DELIVERING HEALTHCARE SERVICES THAT TAKE COMPLICATING FACTORS INTO ACCOUNT

REFERENCE TO A MICROFICHE APPENDIX

An Appendix in the form of 1 microfiche containing a total of 53 frames forms a part of the disclosure herein.

BACKGROUND OF THE INVENTION

This invention relates generally to the management of a healthcare system, and, more specifically, to techniques for estimating charges for treating patients with defined primary and collateral illnesses.

There have been several statistical techniques proposed or implemented that have a goal of homogeneously grouping encounters of patients within the healthcare system by some measure of the outcome of the encounter, such as by the length of stay in a hospital or charges of the healthcare provider to render the healthcare services. Most of this effort has been directed to analyzing the resource consumption of in-patient (hospital) stays. Common to these systems is the categorization of each instance of the delivery of healthcare services into one of a large number, usually hundreds, of categories of illnesses and/or treatments. It is desired that the charges of all services in a given category be quite close to each other in order that an average of such charges can be used as a measure of what all services falling within that category should cost. That is, for example, when a patient is treated for one condition, such as congestive heart failure, an

average of all charges for other patients treated for the same condition is taken as a measure of what the charges should be to treat this specific patient.

The United States government uses such a system of 470+ Diagnosis Related Groups ("DRGs") to reimburse healthcare providers under Medicare for hospital admissions. Many illnesses are defined by multiple DRGs that differ by an age range of the patient or whether there exists a co-morbidity or complication along with the principal diagnosis (the diagnosis which occasioned the admission). But this one separate category for the existence of any co-morbidity or complication does not take into account the large differences in the complexities of illnesses that can result among the large number of secondary or collateral conditions that are possible with any given primary illness. Health providers code diagnoses and procedures performed by use of the International Classification of Diseases - 9th Revision, Clinical Modification ("ICD-9-CM"), approximately 15,000 different codes being in use. Each such code is grouped into individual ones of the DRG's, and a reimbursement amount associated with that DRG is then paid to the hospital or other health provider, no matter how more expensive than normal the treatment may be because of extraordinary secondary illnesses and the like.

It has long been recognized that there is a significant variation in the cost to treat patients within one category, so that the average is not a good predictor of what the charges for treating any particular patient will or should be. Therefore, there has been a significant effort to select categories and/or increase the number of categories to improve the homogeneity of the charges within each category. It has been thought that this is the way to obtain average charges that can be reliably used to estimate what the

charges should be for the purpose of reimbursing the healthcare provider or determining expected charges that can be used to evaluate the efficiency of the healthcare provider. But such techniques have not sufficiently reduced the variation of charges in individual categories to bring about this result. It is not known what portion of the variations are due to differences in the level of illness of the patients and what is caused by differences in the efficiency or style of the healthcare providers. It is the efficiency of the healthcare providers that is desired to be quantified in order to manage them within a healthcare system.

A large body of medical literature documents that patients who are older, have more serious and complex illnesses which extend across multiple body systems (heart, lungs, etc.) are at greater risk of exhibiting higher mortalities, having poorer health and functional status, and consuming greater resources. Therefore, it is a principal object of the present invention to provide a technique of analyzing patients' health data that improves the ability to compare the performance of healthcare providers by significantly reducing variations between expected and actual outcomes (such as charges) due to differences in clinical complexity (severity of illness, and the existence and severity of co-morbid status) among the patients.

It is another principal object of the present invention to provide a technique for improving the accuracy of estimating likely charges (expenditure of resources) for treating a given patient.

It is a further object of the present invention to provide a technique for estimating the financial burden of each illness within each patient in such a way as to allow independent assessments of each illness.

SUMMARY OF THE INVENTION

These and additional objects are accomplished by the present invention, wherein a significant departure has been made in the continuing efforts by others to redefine the categories of illnesses in order to improve their homogeneity. Briefly and generally, the present invention takes a much different approach by applying techniques of regression analysis in particular ways to significantly reduce variations in estimated outcomes of treatment that are caused by the large variation in the level of overall clinical complexities of patients that are being treated for the same primary illnesses and collateral illnesses. Estimates of charges for such treatment are made by quantitatively including the effects of any other illnesses or complicating factors that are revealed by the input data to be specific to a given patient. This significantly reduces patient variability as a cause of differing charges to treat different patients for the same illness. Remaining differences are then primarily the result of differences among health providers, thus allowing their performance to be objectively evaluated and improved.

According to the present invention, a mathematical estimate model is built for each of a list of defined primary illnesses. The outcome of expected charges is expressed as a function of model variables and regression coefficients taken or derivable from data within historic records of patient encounters with health providers. The data upon which the variables and coefficients are dependent include data of secondary illnesses and other complicating factors that affect the charges which are a surrogate for medical resources consumed by the diagnostic and treatment processes ordered by physicians and other care givers. A set of regression coefficients is calculated by applying the mathematical model to a historic database of health

encounter records of a large population of patients. These regression coefficients are stored in a table. An estimate of charges is then made for an individual patient or group of patients by reading from this table the applicable coefficients and using them in the same mathematical model as was used to calculate the coefficients but now with the new patient data.

Since these coefficients and the estimate model include the effect of the specific secondary and collateral illnesses and other complicating factors in the large population database, the estimate takes into account the specific health conditions of the patient or group of patients that can affect the amount of resources which will be expended to treat the primary illness. This is much more accurate than merely averaging the charges for all patients having the same primary illness, as has been done before, even when two or more categories of the primary illness are maintained according to the complicating or co-morbid conditions, as is done with the DRG and other software groupers. According to the present invention, estimates are made directly from the data without going through some intermediate classification based upon clinical complexity (such as illness severity).

The present invention also provides the ability to analyze secondary (complicating) and collateral (co-morbid) illnesses independent of all other illnesses. This allows physicians to understand which illness and its diagnosis and treatment resource utilization accounts for more or less of the observed charges (spending). This is not possible in a system which uses averaging and thereby loses the specificity of each illness and its incremental impact on the observed total charges or resources consumed.

According to one specific aspect of the present invention, the regression analysis is performed

in two or more stages using the estimates resulting from a previous stage as independent variables in one or more subsequent stages, both when calculating the set of regression coefficients and when using them to make an estimate for a specific patient or group of patients. That is, two or more estimate models are used, the first providing a rough estimate of charges which is then used as a variable of the second model. This technique reduces the number of quantities in each of the two or more mathematical models, which makes the processing more manageable.

According to another specific aspect of the present invention, two or more similar but different mathematical models are used in each estimate stage. One model uses all the variables believed to provide the best estimate for that stage but in case there is not enough data of all those variables, one or more additional models are provided with fewer variables or variables based upon patient data that is more likely to occur for most of the primary illnesses.

The foregoing inventive processing and charge estimating techniques are useful with in-patient data alone, some specific set of out-patient data, or some sub-set of these. However, the techniques of the present invention are most useful, although not limited, to the management of healthcare systems when data of the full continuum of care is used. This allows calculating all charges expected to be incurred in connection with an illness in any care setting over time. Therefore, it is preferred to form summary records from data of records of encounters of patients with both in-patient and all types of out-patient healthcare. Estimates specific to individual patients or a group of patients are then made, according to another aspect of the present invention, for all charges related to a primary

or collateral illness, or for charges related to specific components of provided services.

In a specific implementation of the present invention, patient data is maintained in four categories, generally according to the setting in which healthcare service is provided. One category is in-patient ("IP") services provided while the patient is admitted to a hospital. Another is a visit to a doctor's office ("OF"). A third category is a day encounter ("DE"), which includes one day visits to a medical facility for a procedure. The last category is therapeutic series ("TS"), which includes a closely related series of encounters such as radiation treatments, chemotherapy, and the like. Patient data is obtained from encounter records including hospital discharge forms and insurance reimbursement forms.

As an early step of the processing, the encounter records are grouped by episodes of care. Each episode is one day in length, except for extended treatments resulting from a hospitalization or a therapeutic series. One of a list of primary illnesses is identified for each episode from the data of the encounter record(s) that make(s) up the episode. Such records nearly always indicate a primary diagnosis of the patient's condition, indicating the reason for the encounter, which is the most important piece of information which is used to determine the primary illness. However, the primary illness for a given episode is determined by an algorithm that considers whether an illness that would otherwise be indicated is really a continuation or recurrence of a previous illness of the patient. Any collateral and secondary illnesses (sub-illnesses) indicated by the data are also carried as part of the episode records since this is important to estimating charges, as mentioned above. The encounter records often indicate secondary diagnoses

that are used to determine such sub-illnesses but data of prior episodes can also be used.

Expected charges for an episode can be calculated by the present invention for the purpose of comparing the performance of providers of the same episode services to different patients. Such episodes are of a single type of service, such as IP, DE or TS. But it is often preferable to be able to estimate the charges to manage a patient's entire illness which usually includes several episodes of care. If an illness is chronic, such as diabetes mellitus, it has an indefinite duration and the charges are estimated per year. If an illness is acute, such as a broken arm, it has a finite duration and such a duration is assigned to each type of such occurrence. Episodes of an acute illness are then included in a particular occurrence of that illness so long as they are within the specified duration of the first episode within this occurrence. Episodes falling outside of that window usually cause the beginning of a new occurrence of the same illness. The overall efficiency of health providers in treating a chronic illness ("illness") and an acute illness ("illness occurrence") can then be compared. Other combinations can also be estimated by the present invention by limiting the types of episodes included in each illness and illness occurrence, such as using only OF and DE.

It is also often desired to be able to estimate charges for particular categories or classes of care used to treat an illness or illness occurrence. According to a further aspect of the present invention, a list of procedure classes is established, such as emergency room visits, and radiology procedures. Charges are estimated for each such procedure class for a given illness from the data maintained as part of the illness or illness occurrence records. This allows

comparison among health providers as to which are using the emergency room too much or two little, or sending patients for radiology examinations too much or too little, and so forth.

Although the present invention is primarily described herein with respect to the example of estimating charges, the various aspects of the present invention are also applicable to estimating other outcomes of treatment. The length of stay in a hospital, mortality, patient satisfaction and a measure of overall patient health status are examples of other such outcomes.

Additional objects, features, and advantages of the various aspects of the present invention are included in the following description of its preferred embodiments, which description should be taken in conjunction with the accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

The charts of Figures 1A-C illustrate the relative variations of components of healthcare before and after the present invention;

Figure 2 shows the major stages of the data processing used to implement the present invention;

Figure 3 outlines in a general way the procedures used to estimate healthcare charges;

Figure 4 shows an example of the illness decomposition processing for a single patient with multiple simultaneous illnesses;

Figures 5A-B illustrate two time durations used in the illness decomposition portion of the present invention;

Figure 6 is a general flow chart of the processing of patient data in order to form a table of regression coefficients;

Figure 7 is a general flow chart of the processing used to estimate charges;

Figure 8 is a flow chart illustrating the use of multiple alternative estimate models in the processing of either Figure 6 or 7;

Figure 9 illustrates, in block diagram form, a typical computer system used to carry out the processing illustrated in Figures 2-8; and

Figure 10 schematically shows utilization of a memory of the computer system of Figure 9.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

The three bar charts of Figure 1 illustrate a primary goal of the present invention, compared with the effects of the current direction of healthcare management. Referring first to Figure 1A, three components of variations in charges by healthcare providers for treating a particular primary illness among a population of patients are shown. One component 11 shows a theoretical proportion of the variation in charges that is inherent in a patient population. These charge variations are required to diagnose and treat the differences in clinical complexities of the patients in the group. Some patients are more clinically complex than others and therefore require a greater expenditure of healthcare resources. One patient may have a primary illness plus a secondary or collateral illness which causes the complexity and appropriate resource consumption to be greater. Another patient with the same age and primary illness that has no such other secondary or collateral condition will cost less to manage.

A component 13 of the bar chart of Figure 1A indicates the portion of variation in charges that is due to differences in the operation of hospitals, clinics, laboratories and other institutions. One hospital, for example, may have patients who remain longer than in another hospital because of inefficiencies in discharge procedures, thus incurring

greater charges for treating the same illness. A component 15 represents the variation in charges due to physicians. Some physicians order more laboratory tests, radiology, and the like, or require patients to return for more office visits, than others. The variations represented by the components 13 and 15 are desired to be minimized by effective management of healthcare institutions and physicians.

The bar chart of Figure 1B illustrates a result of using the data processing techniques of the present invention. A component 11' of variations of charges to treat the group of patients with the same illness remains unchanged from the component of Figure 1A. Indeed, this must be the case since variations in clinical complexities of the illnesses among the population of patients cannot be changed by statistical manipulation, and sicker patients cost more to treat properly. What can be controlled, without failing to give the sicker patients the care they need, is the variation among the institutions and physicians, as indicated by components 13' and 15' that are reduced versions of the variation components 13 and 15 of Figure 1A. That is, the techniques of the present invention allow, for example, the identification of inefficient care processes and physicians who order too many laboratory tests, or not enough, when treating the same illness, after taking into account the complexity of the illnesses of the physicians' patients. This then allows management of the healthcare providers by establishing norms so physicians and institutions can improve care processes which caused their deviations from the norm for each of a large number of defined illnesses. The present invention allows the physician and institution variation components 13 and 15 to be identified and therefore appropriately reduced, as shown in Figure 1B, as opposed to previous techniques that result in cost

reductions that inappropriately reduce expenditures in the patient component 11, as shown in Figure 1C.

The present invention provides an estimate of treating a particular patient, or group of patients, for a specific illness, or group of illnesses, that accurately accounts for differences in resource consumption (charges) due to varying levels of clinical complexity of the patients. As described below, this is done by forming an indexed data set from healthcare records of a large (at least several thousand) population of patients. A large table of linear regression coefficients is calculated from this indexed data set, one set of regression coefficients for each of several hundred defined illnesses, that takes into account related illnesses (co-morbidities), complications and other complicating factors. To form an estimate of charges to treat a particular patient for a given illness, the coefficients for the same illness are read from the table of regression coefficients and used in the same estimate model that was used to calculate them from the indexed data set. The resulting estimate will have small variations from the particular level of sickness of the patient (component 11') since that is taken into account. Any significant difference between such an estimate and the actual charges can then be attributed to the healthcare providers. This difference is valuable information that is then used to advise or manage the care processes of healthcare providers, resulting in the small components 13' and 15' of variation that are attributed to the providers.

Without the ability to identify the causes of variations in the costs needed to appropriately treat different patients with similar illnesses, the previous systems tend to reduce variations by considering that treatment of all patients with the same illness should cost about the same. The only exceptions to this

include providing one or two separate sub-categories associated within a given illness for those patients who are elderly, have any co-morbidity or complication, and the like, based upon the resulting added cost to treat such patients. But this simply provides an average cost to treat all patients for a given illness, or perhaps one or two additional average costs for older patients and/or those who are sicker from some other illness. The added category for patients with a co-morbidity cannot take into account the wide spread in the amount of additional cost incurred to treat those patients having different one(s) of hundreds of possible secondary illnesses.

Figure 1C illustrates a highly undesirable outcome of managed care initiatives which is the result of not having the capabilities of the present invention. The inappropriate racheting down of healthcare costs is occurring because the present health data is insufficiently risk adjusted and therefore unreliable. Payers and governmental agencies gather these unadjusted data and use them for the purpose of reimbursing, managing and evaluating healthcare providers. The limitation of this approach is that patients' clinical complexities are unquantified and therefore the appropriate numbers and types of treatment resources are unknown. This penalizes physicians and hospitals who manage the most difficult cases and ultimately withholds care from the neediest patients. As this illustration and recent history clearly demonstrate, when fewer total dollars are allocated for care, the variations of physicians and hospitals remain virtually the same, and the costs necessary to manage patients' clinical variations come largely out of the patient component in the form of withholding of care.

Thus, as shown in Figure 1C, the total variation of estimates to treat a given illness will

likely be reduced from the picture of Figure 1A of the way it used to be. But this reduction in total variation is also causing an artificial reduction in a patient component 11'', as well as in institution and physician components 13'' and 15''. The reduction in the patient component 11'' can only mean that the sicker patients are not receiving the care they need, and/or those not so sick are receiving more care than they need. Adequate information is not being provided to healthcare providers from which they can improve their care processes. The present invention rectifies this fundamental deficiency.

Referring to Figure 2, the stages of data processing used to implement the present invention are outlined. As an input to the processing indicated at 17, data is provided of encounter records for a patient or patients whose expected charges are to be estimated. Data from these records are input into the computer system. These patient records include hospital discharge forms, insurance reimbursement forms, and similar sources of patient data. Data on these forms include identifying information of the patient, including gender and age, codes of a primary and any secondary diagnoses, codes of any procedures performed, both primary and secondary, any applicable DRGs, date(s) the services were provided, identifying information of the health provider and charges for providing the services.

The remaining stages of Figure 2 are generally outlined, with added explanation being provided below. Input data 17 is decomposed in processing of a stage 19. The primary purpose of the stage 19 is to group the encounter records into episodes of care for identified primary and collateral illnesses. These results are indicated as an output 21. A next stage 23 estimates charges for various of these episodes and complete

illnesses, adding the results of this processing to the output 21 of the decomposition stage 19, forming a more complete output 25. The processing can stop here but it is often desirable to include another processing stage 27 to calculate, from the results of the output 25, charges for various types of procedures. An output 29 includes the results of each of the processing stages 19, 23 and 27. These results include estimated charges for the patient or group of patients that can be compared with the actual charges or otherwise used in the management of a healthcare system.

Before explaining the steps of the estimation processing of Figure 2 in more detail, reference is made to Figure 3 wherein the same estimation processing is illustrated along with steps to form the table of regression coefficients upon which the estimations are dependent. The table is calculated from the indexed data set of healthcare encounter records of a large population of patients, such as exist with a large health insurance company, health plan of a large corporation, and similarly other sources. Generally, the larger the number of patients and the longer the period of time over which the encounters extend, the better. Data from such records is input into the computer in a step 31, followed by an illness decomposition step 33 and an estimating step 35. The algorithms used in the steps 33 and 35 are essentially the same as those of the steps 19 and 23 (and also preferably 27) of Figure 2. Using the estimate models, regression analysis, such as least squares analysis, is used to calculate the regression coefficients. The result is a table 37 of regression coefficients for these estimate models. This table can be regularly updated by repeating the processing of the steps 33 and 35 on an enlarged volume of encounter records 31 that occurs over time.

When making an estimate of charges to treat a specific patient or group of patients (who will usually not be included in the population from which the encounter records 31 are taken), steps 39, 41 and 43 of Figure 3 are performed, which correspond, respectively, to those of blocks 17, 19 and 23 (and also possibly 27) of the diagram of Figure 2. In the estimation step 43 (corresponding to stages 23 and 27 of Figure 2), appropriate ones of the regression coefficients are drawn from the database 37 for the primary illness(s) whose expected charges to treat are being estimated.

Returning to the estimation illustration of Figure 2, an early step in the decomposition stage 19 is to identify a primary illness (denoted IL0) for each of the encounter records. This is done primarily from the diagnoses and procedure codes of the individual encounter records. ICD-9 or other types of codes, including DRGs if that is all that is shown on a record, are mapped into individual categories of an illness table. Up to nine secondary illnesses (IL1-IL9) are also identified for each encounter record, primarily from secondary diagnoses indicated on the patient record. These secondary illnesses, or "sub-illnesses", are primarily used later in the charge estimation stage, where their inclusion allows an accurate estimate to be made as to the risk of incurring expenses for an individual patient.

Another step in the decomposition stage is to gather the encounter records into records of episodes of care. Encounter records are grouped together according to certain rules concerning the duration of an individual episode. A primary illness and set of collateral illnesses are associated with each episode. For in-patient services (IP), a continuous stay in the hospital is considered to be a single episode. For office visits (OF), an episode is one day in length, as

is a day encounter (DE) episode. A therapeutic series (TS) episode has a single length extending from the first to the last of a number of visits for therapy. If two or more episodes of different types that are created by these rules occur on the same day for the same illness, they are combined into a single episode that is given the type of the record having the highest priority. The episode types, in order of priority, are IP, TS, DE and OF. This usually results in the records having the lesser amount of charges being, in effect, folded into the one having the greater amount of charges.

In assigning a primary illness to the episode records, the history of the patient for which the estimate is being made is utilized. The illness table makes such history relevant for certain of the illnesses mapped from the encounter records. An example of this is illustrated by Figure 5A, wherein an episode is initially identified from the data on its encounter record(s) to be congestive heart failure. This is one of the illnesses in the illness table that is coded to be an illness itself, or an indication of some other illness, such as, in this example, coronary artery disease. For this type of initially mapped illness, the processing looks back in the same patient's records in order to determine whether an episode 49 of the higher level disease, in this case coronary artery disease, occurred within a time "t" before the current episode being evaluated. If so, the current episode is reclassified from the initially mapped illness to the higher level illness, in this case coronary artery disease. If not, the illness identified for the current episode remains that initially determined, in this case congestive heart failure.

In a specific form of the present invention, it is contemplated that estimates will be made for

individual episodes of each type of care, namely IP, OF, DE and TS. Although this provides very useful information for managing the delivery of healthcare services, it has been found to be of even greater help to group episodes of care for the same primary illness over the length of that illness. In the case of acute illnesses, a broken arm being an example, the episodes of care extend over a predictable period of time. It is the cost of treating that entire occurrence of an acute illness that is useful to estimate for the purpose of comparison with the actual charges of the health providers. In this way, the delivery of services (care processes) to treat each illness across the entire continuum of care can be managed by the providers using process improvement techniques.

Each of the illnesses in the illness table that can be of an acute type is specified to last a certain time duration that is determined from experience for that illness. Such a duration "t1" is indicated for an illness occurrence 51 of Figure 5B. This illness occurrence is shown to include several episodes of care 53, 55 and 57. The specified duration t1 commences with the first episode 53 for its primary illness. Any episodes occurring after time t1 will not be considered part of the same illness occurrence 51 but rather will begin a new one. An exception to this is in rather infrequent cases where an episode of that same illness has begun before the end of the period t1, in which case the duration of the illness occurrence is extended until the end of that episode. An example of this is shown in Figure 5B, where another episode 59 of the same primary illness begins on the last day of the duration t1. The result is to extend the duration of this illness occurrence by a time "t2".

In addition to estimating charges for acute illnesses, charges are also estimated for chronic

illnesses, diabetes being an example. Since chronic illnesses do not have a defined duration, but rather are indefinite in length, the cost is estimated per some unit of time, such as dollars per year, for providing care to a particular patient because of the chronic illness. By so estimating, a very useful comparison is made as to how various health providers take care of such illnesses.

Estimates of providing care for illness occurrences (acute) and illnesses (chronic) is preferably made of a combination of all forms of care, IP, OF, DE and TS. It has also been found useful, however, to make these estimates with only episodes of office visits (OF) and day encounters (DE). This reflects those services for which a primary care physician is usually responsible. Thus, the performance of primary care physicians can best be ascertained by such more limited illness and illness occurrence charge estimates.

An example of a single patient with multiple illnesses occurring at the same time is given in Figure 4. Over the two year period shown, this hypothetical patient has two chronic illnesses, asthma 61 and prostate cancer 63. Treatment of asthma includes two office visits 67 and 69 for one flare-up of the illness. A subsequent flare-up of the asthma requires three office visits 71, 73 and 75, plus a hospitalization 77. Two post-hospitalization office visits 79 and 81 follow. The prostate cancer requires an office visit 83, a needle biopsy (day encounter) 85, and routine care office visits 89, 91, 95 and 97. A therapeutic series 93 provides radiation treatment for the cancer. Each of the boxes within the illnesses 61 and 63 is an episode of care for the respective illnesses. The therapeutic series episode 93 extends over some period of time during which the two office episodes 89 and 91 occur.

If an office visit occurs on the same day as a therapeutic series visit or treatment, charges for this office visit will be included in the therapeutic series episode 93.

Further, during this same period the patient has an acute illness 99 of a broken ankle. This illness occurrence commences with an office visit 101, followed immediately by a day encounter 103 to place a cast on the ankle. Two follow-up office visits 105 and 107 end this occurrence of the illness, which has a finite duration defined in the illness table. Indeed, there is no further activity until much later when another office visit 109 takes place for a broken ankle again. This begins a second occurrence 111 of the same illness. Because the initial encounter 109 occurs after the close of the first illness occurrence 99, a new illness occurrence 111 is begun with this office visit. The initial office visit 109 is followed by a day encounter 113 and a follow-up office visit 115.

After completion of the decomposition stage 19 (Figure 2), the patient encounter records have been decomposed into episodes, illnesses and illness occurrences. The present invention provides for estimating the charges for treating each of the illnesses 61, 63, 99 and 111, either with all the episodes of care shown or only with various combinations of them. This later option gives different results in each of the chronic illnesses 61 and 63 since the hospitalization and therapeutic series would not be included in the estimate.

The estimator stages 23 (Figure 2), 35 and 43 (Figure 3) use hierarchical linear regression analysis to estimate charges. A flowchart of Figure 6 illustrates the processing of step 35 to generate the table of regression coefficients that are used in calculating expected charges. In a first step 121, an

estimate model 1 is built by setting an initial rough estimate of charges EXP_CH0 equal to a sum of mathematical terms of variables and regression coefficients, such as,

$$EXP\_CH0 = a_0 + a_1x_1 + a_2x_2 + . . . \quad (1)$$

where x_1 and x_2 are model variables, and a_0, a_1 and a_2 are regression coefficients. The model variables are taken from the data of the large population of patients, the number of diagnoses (numdx) being an example of one variable "x". The EXP_CH0 is set to the actual charges incurred and the coefficients "a" are calculated for each value of one, or a combination of two or more, grouping variables taken from the patient encounter record data. This calculation is made by use of the least squares algorithm in order to find the coefficients "a" that cause equation (1) above to have the best fit with the data. An example of grouping variables is a combination of a primary diagnosis (dx0) and primary procedure (pr0) reported on the encounter records. That is, the coefficients "a" of equation (1) are calculated for each combination of "pr0" and "dx0" in the data of the patient population.

Instead of using a single estimate model with all the desired variables "x", and then solve for its many regression coefficients "a" for rather complicated numbers of combinations of grouping variables, two or more estimate models are used in order to make the processing easier and avoid having to simplify the estimate model to eliminate terms that are believed to be important to the estimate. Indeed, in one embodiment, equation (1) is reduced to only the first two terms. After the coefficients "a" of equation (1) are calculated in the manner described, that equation is used to calculate EXP_CH0 from the indexed patient data set. An estimate model 2 is then built, as indicated by

a step 123 of Figure 6, by using EXP_CH0 as a variable of the estimate model 2,

$$EXP\_CH1 = b_0 + b_1 (EXP\_CH0) + b_2 y_2 + b_3 y_3 + \ldots \quad (2)$$

where the regression coefficients are denoted by "b" and other variables by "y". The coefficients of this model 2 are solved by the least squares algorithm for each value of a grouping variable, or combination of two or more grouping variables, by setting EXP_CH1 equal to the actual charges. The calculated values of the coefficients "b" are then substituted back into equation (2) and EXP_CH1 is calculated for use in a third estimate model 125. Use of the estimate EXP_CH0, calculated by equation (1), as a variable in equation (2) makes the technique hierarchical.

Estimate model 3 solves for charge differences, or "deltas", for each sub-illness present in the patient data, thus directly correlating the estimates for treating a given primary illness with the concurrent existence of specific collateral illnesses (sub-illnesses). The estimate model 3 is,

$$(EXP\_CH1 - \text{Actual Charges}) = c_0 + c_1 z_1 + c_2 z_2 + \ldots \quad (3)$$

where the regression coefficients are denoted by "c" and model variables by "z". The coefficients are calculated by use of equation (3) for each collateral illness. Thereafter, the coefficients are substituted back into equation (3) and delta charges (EXP_CH1 - Actual Charges) are calculated.

A next and final step 127 of the regression analysis uses a linear regression equation (4) that equates a final estimate EXP_CH2 to a series of coefficients and variables. Various averages of the delta charges calculated in step 125 are used as model variables in equation (4). EXP_CH2 is set equal to the actual charges, and the coefficients of equation (4) calculated by least squares.

As a final step 129 of the flow chart of Figure 6, all of the regression coefficients from steps 121, 123, 125 and 127 are stored in a table within the computer mass storage memory. This table is necessarily quite large since different values of the many coefficients of equations (1) - (4) have been determined for different values of grouping variables taken from the patient records. Also, there are a set of such coefficients by primary illness for each episode, illness and illness occurrence.

Referring to the processing flow chart of Figure 7, a first step 131 of determining expected charges for a given episode, illness or illness occurrence, as desired, for a given patient is to read from the coefficient table in memory those coefficients that are appropriate for the patient data. Since these are the coefficients used in each of the estimate models, they can also be read in conjunction with the use of those models.

As indicated by a step 133, the estimate model equation (1) is solved for EXP_CH0. Data of the given patient provide the model variables "x" of this equation, and a set of coefficients "a" is taken from the table formed in the step 129 as recited by the patient data. For example, if the coefficients were determined in the step 121 for each combination of a primary diagnosis (dx0) and primary procedure (pr0), then the coefficients determined for the specific combination of dx0 and pr0 existing in the subject patient's data are read from the stored table and substituted into equation (1).

Once EXP_CH0 is calculated, a next step 135 solves the estimate model equation (2) for EXP_CH1 by substituting actual data for the model variables "y" and choosing the coefficients "b" from the coefficient table that were determined for the specific grouping variables

used in the step 123. Similarly, estimate model equations (3) and (4) are solved in respective steps 137 and 139. An expected charge EXP_CH2 is the result. This charge estimate is for an episode of care, illness, or illness occurrence, consistent with whether the patient data used and coefficients selected are for an episode, illness or illness occurrence.

It is desired to have the estimate models depend upon as much patient data as is available to provide the best results. But some patient records will not have some of the data that provides the best results. Rather than building a single estimate model for each of the models 1-4 that depends upon the least amount of patient data that is likely to be available most of the time, two or more alternative models are used for individual ones of the models 1-4. One of these models is made dependent upon data that gives the best results but may not always be available in sufficient quantities. A second of these models is made dependent upon a reduced amount or different patient data that is usually always available. The processing of each estimate model then uses the best of the two models for which patient data is available. This has the advantage of providing more accurate estimates than are possible with the patient data that is always available.

As an example, three alternative versions of the model equation (1) each have the number of diagnoses "numdx" as the variable x_1 but the regression analysis is performed in step 121 for three sets of grouping variables to result in three sets of coefficients that are stored in the table. The three sets of grouping variables are, in this example, a combination of the primary diagnosis (dx0) and primary procedure (pr0) for a patient, if they exist, a combination of pr0 and DRG, and the DRG alone. Use of the coefficients determined

for each DRG alone to calculate EXP_CH0 gives results that are not as good as when one of the other versions of the model is used but the technique of using alternative models allows the most accurate result that is possible from the available data.

Figure 8 illustrates the implementation of any of the steps 131, 133, 135 or 137 of Figure 7 where two alternative versions of the estimate model are provided. In a step 141, a version of the model requiring the most data is recalled and a determination made in steps 143 and 145 whether there are coefficients in the table and enough data in the current patient record to use this version. If either the coefficients or the patient data is not available, a second version of the estimate model requiring less data is recalled, in a step 147, and the same tests of steps 143 and 145 made. The first one to pass the tests of the steps 143 and 145 becomes selected for use, as indicated in a step 149. Although use of only two alternative versions of an estimate model is shown, three or more can be employed if there is some advantage in doing so.

An example of a computer system that may be used to carry out the foregoing processing is shown generally in Figures 9 and 10. Figure 9 is a block diagram of the computer system hardware and Figure 10 schematically shows a memory space within the computer system for storing various data files and tables. The hardware includes several functional units that communicate with each other over a common system bus 161. These units include a central processing unit (CPU) 163, a non-volatile read-only-memory (ROM) 165, a volatile random-access-memory (RAM) 167, and a magnetic disk drive mass data storage system 169. Also typically connected to the system bus 161 is a communications unit 171 that includes a modem and/or network interface to a circuit 173 that is a telephone line and/or a computer

network connection. Another input/output unit 175 provides an interface between the bus 161 and at least two circuits 177 and 179 for connection with a keyboard, mouse, monitor, and other standard computer peripheral devices.

Several data files and tables are stored within the disk system 169 for reference by the CPU 163 during execution of various portions of the algorithm described herein. Some of the more important of these files and tables are shown in Figure 10. Separate files are maintained for raw patient data in substantially the form received. Files 181 and 183 respectively store data from hospital discharge forms, such as UB92, and out patient charge records, such as HCFA 1500, and there will generally be several more in yet different formats. This raw input data is taken from these files, as part of the processing, and placed into a common file 185 in a common indexed format. It is the indexed data file 185 that is the source of patient data throughout the remaining processing. The file 185 is updated as the patient input data changes.

Two static tables 187 and 189 are utilized, but there can be more. The table 187 identifies, for medical code data from the indexed file 185, the type of the individual episodes of care. The table 189 defines illnesses and sub-illnesses for patient data from the file 185.

Another file 191 includes the calculated regression coefficients, so can change from time-to-time as the amount of patient data changes and at least some of the coefficients are recalculated. This file is accessed for individual ones of the regression coefficients as needed during the processing. A final file 193 illustrated in Figure 10 stores the resultant calculated estimated charges. Additional files and

tables can also be included as part of the processing system shown.

An example of an algorithm to implement the processing described herein is provided in a microfiche Appendix that is being filed with this application and forms a part of this description. The data files and tables of Figure 10 are utilized in that algorithm.

Although the present invention has been described with respect to its preferred embodiments, it will be understood that it is entitled to protection within the full scope of the appended claims.

IT IS CLAIMED:

1. A method of managing delivery of services by healthcare providers to medical patients, comprising:

(A) creating a table of regression coefficients from individual records of encounters of a population of patients with healthcare providers, by a method including:

accumulating and storing data from the encounter records in a mass storage system of a computer, data of individual ones of the encounter records including at least (1) an identity of a single patient, (2) charges of the healthcare providers for the encounter, and (3) at least one diagnosis made or procedure performed,

grouping said encounter records, from information provided therein, into a plurality of summary records for individual ones of the population of patients and one of a plurality of primary illnesses,

establishing an estimate model of a total amount of charges for the encounters within a summary record as a function of a plurality of model variables and regression coefficients taken or derivable from the data within said summary records,

solving, separately for individual ones of the primary illnesses, the estimate model for the regression coefficients that optimizes fits of said estimate model with the data within said summary records, and

storing said regression coefficients in a table within the computer mass storage system, and

(B) estimating the charges for treating an illness of at least one patient, by a method including:

grouping, within the computer, data of the encounter records of said at least one patient, from information provided therein, into at least one summary record for one of the plurality of primary illnesses,

reading the regression coefficients from said stored table for the primary illness of said summary record,

solving the estimate model for estimated charges by use of the read regression coefficients, and

(C) utilizing the estimated charges to manage the delivery of health services by the healthcare providers.

2. The method according to claim 1, wherein accumulating and storing records of individual encounters includes accumulating and storing records of hospital and outpatient encounters for individual patients, and wherein grouping the encounter records includes grouping records of hospital and outpatient encounters into common ones of the summary records.

3. The method according to claim 2, wherein the outpatient encounters for which data is accumulated, stored and grouped include office visits, day encounters and therapeutic services.

4. The method according to claim 1, wherein data of individual ones of the encounter records include data of patient conditions that are collateral to the primary illnesses, and wherein one or more of the model variables and/or regression coefficients of the estimate

model is taken or derived from data of the collateral conditions.

5. The method according to claim 4, wherein data of collateral conditions includes data of illnesses other than the primary illness.

6. The method according to claim 1, wherein grouping said encounter records includes determining a primary illness for individual ones of the summary records.

7. The method according to claim 6, wherein determining a primary illness includes reviewing summary record data for an individual patient for data of encounters occurring prior to those for which data are included in the summary record.

8. The method according to claim 1, wherein establishing the estimate model includes establishing more than one specific estimate model with an estimate of charges of one specific estimate model being used as a model variable of a second estimate model, and wherein solving the estimate model includes solving the specific estimate models in sequence.

9. The method according to claim 8, wherein said one specific estimate model is chosen from multiple alternative specific estimate models that use a different set of model variables and/or regression coefficients from each other, and wherein solving the specific estimate model includes solving one of the multiple estimate models having the greater number of variables and/or regression coefficients for which encounter record data is available.

10. The method according to claim 1, wherein a length of time for which data of encounters is included in one of the summary records is specified for the primary illness of the summary record.

11. The method according to claim 10, wherein said length of time is an indefinite duration for chronic ones of the primary illnesses.

12. The method according to claim 10, wherein said length of time is a specified finite duration for acute ones of the primary illnesses.

13. The method according to claim 12, wherein said specified finite duration of time is extended when data exists of a succession of related encounters that begin within the specified duration but extend beyond said specified duration.

14. The method according to claim 1, wherein grouping data of the encounter records into the summary records includes first grouping such data into episodes of one of the plurality of primary illnesses and then grouping the data of the episodes into the summary records of the same primary illnesses.

15. The method according to claim 1, wherein the data of the encounter records of said at least one individual patient are not within the data from encounter records used to create the table of regression coefficients.

16. The method according to claim 1, wherein creating the table of regression coefficients and estimating the charges for treating an illness of at least one patient each include determining, from the

summary records data and estimated charges, charges for one of a plurality of specific procedures performed.

17. A method of estimating charges of healthcare providers to at least one patient for the purpose of providing advice on the efficiency of such providers, comprising:

accumulating and storing, in a mass storage system of a computer, data from records of encounters of said at least one patient with healthcare providers that includes at least (1) an identity of said at least one patient, (2) charges of the healthcare providers for the encounter, and (3) at least one diagnosis made or procedure performed,

grouping data of said estimate encounter records, from information provided therein, into at least one summary record of said at least one patient for one of a plurality of primary illnesses,

solving an estimate model of a total amount of charges for the encounters within a summary record as a function of a plurality of model variables and regression coefficients taken or derivable from the data within said at least one summary record, using regression coefficients previously determined with the same estimate model to optimize a fit of said estimate model for a population of patients with data within a summary record corresponding to said at least one summary record, and

utilizing the estimated charges to advise on the efficiency of the healthcare providers in the delivery of health services.

18. A database stored in a mass storage system, comprising:

a plurality of raw data files of individual records in multiple different formats of patient

encounters with both in patient and out patient healthcare providers,

a common patient data file containing an indexed version of the raw data contained in the patient encounter data files,

a plurality of tables containing definitions including those of various illnesses,

a table of regression coefficients calculated from at least the indexed patient data file and the definition tables for various combinations of primary and collateral illnesses, and

an output table containing data of estimated charges that have been calculated for specific patients from healthcare encounter data of such specific patients and at least the table of regression coefficients.

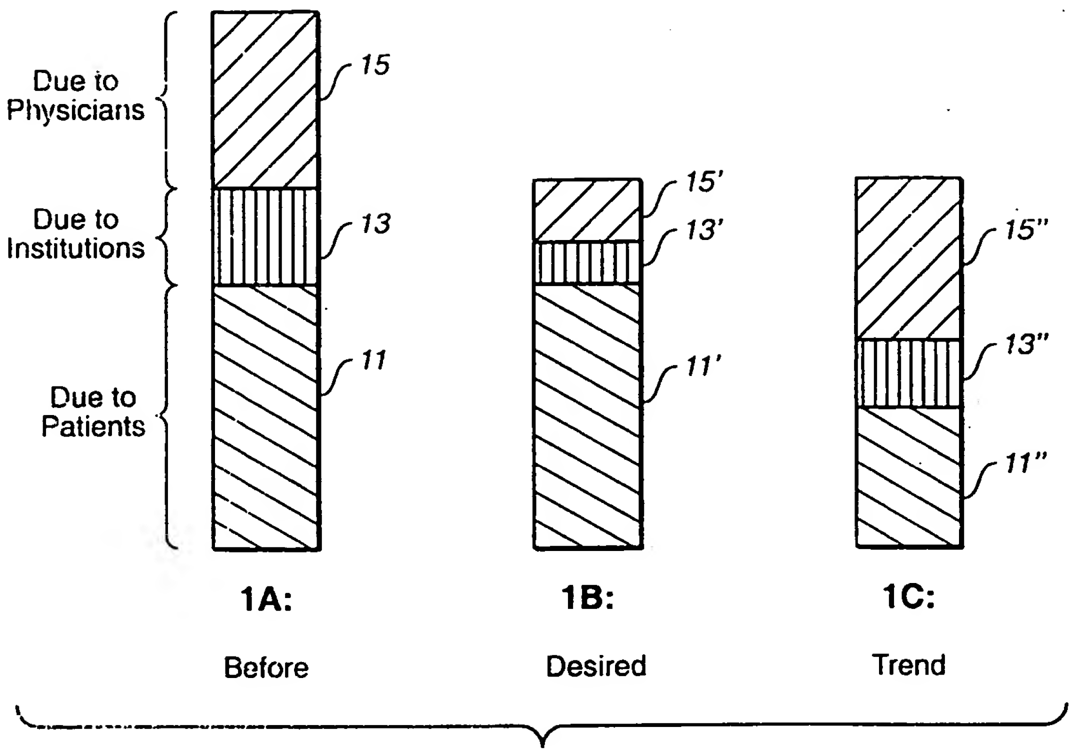
Due to
Physicians
Due to
Institutions
Due to
Patients
15
13
11
15'
13'
11'
15"
13"
11"
1A:
Before
1B:
Desired
1C:
Trend

FIG._1

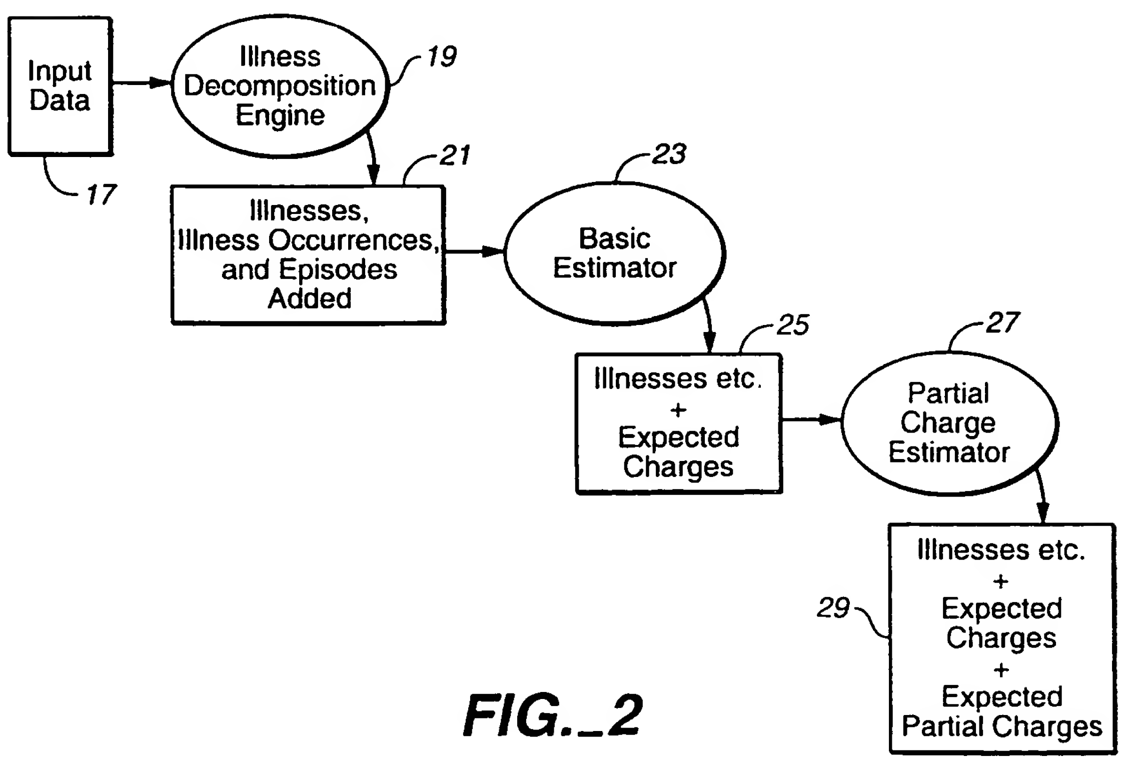
Input
Data
17
Illness
Decomposition
Engine
19
Illnesses,
Illness Occurrences,
and Episodes
Added
21
Basic
Estimator
23
Illnesses etc.
+
Expected
Charges
25
Partial
Charge
Estimator
27
Illnesses etc.
+
Expected
Charges
+
Expected
Partial Charges
29

FIG._2

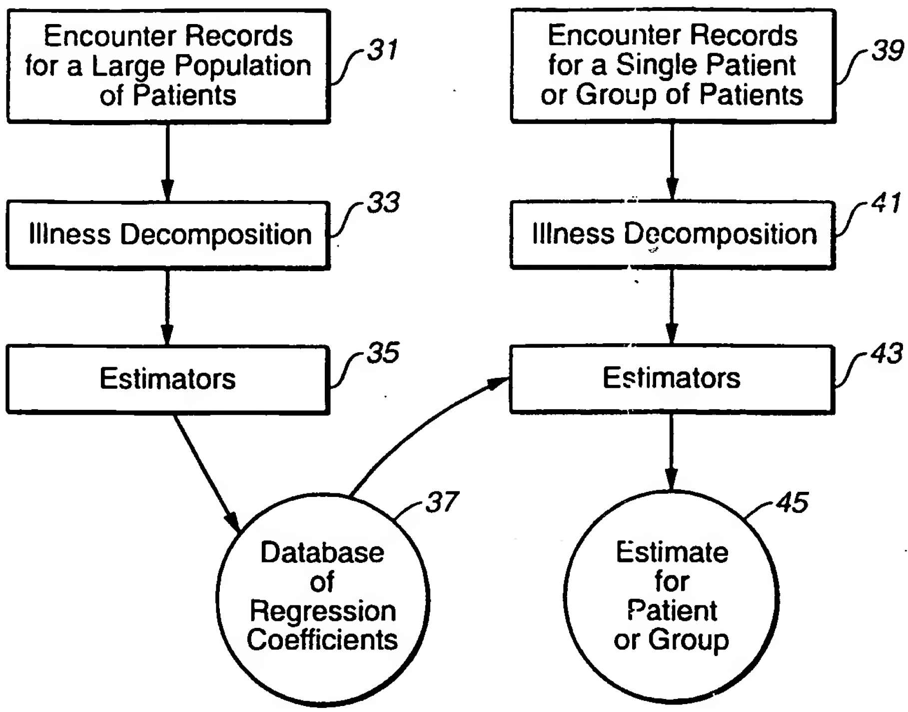

FIG._3

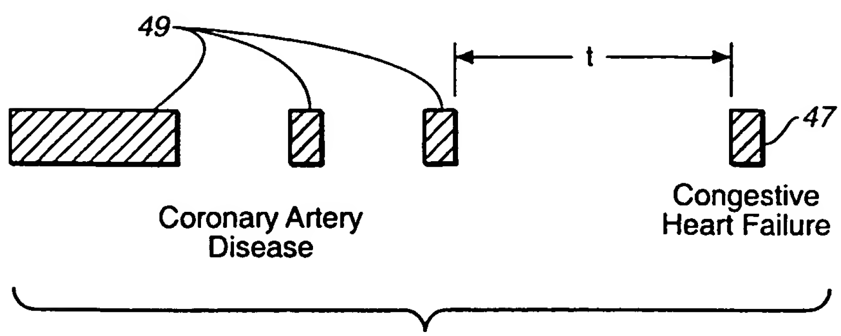

FIG._5A

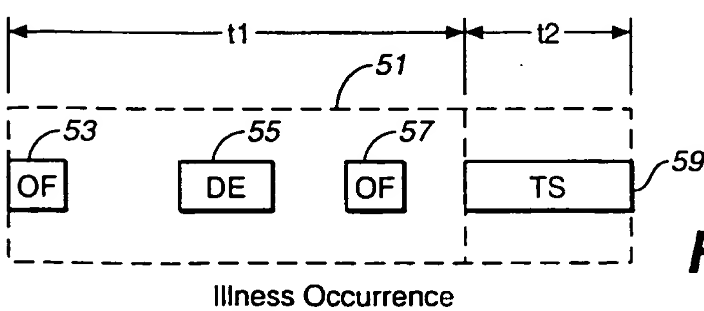

FIG._5B

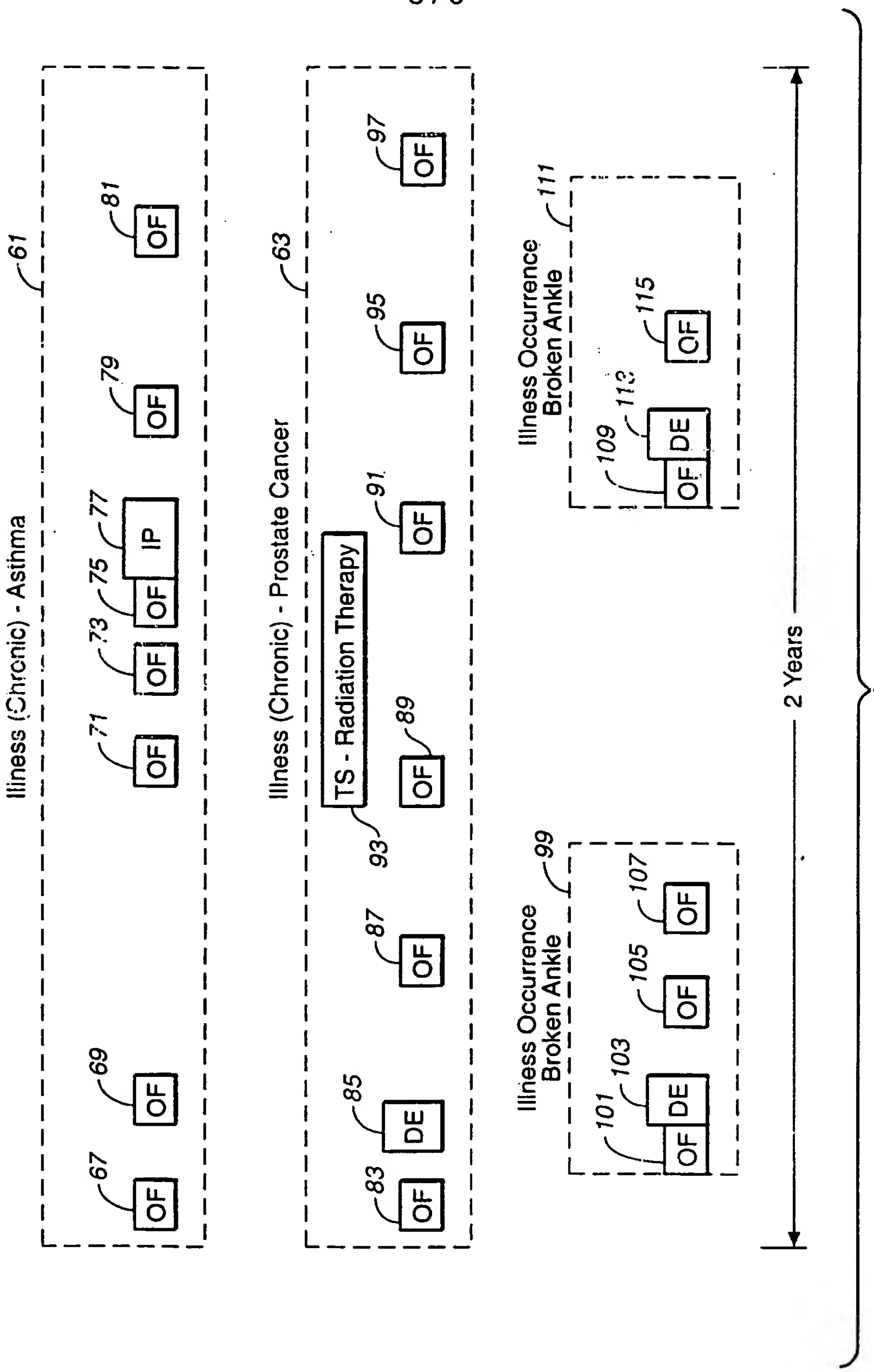
Illness (Chronic) - Asthma
61
OF
67
OF
69
OF
71
OF
73
OF
75
IP
77
OF
79
OF
81
Illness (Chronic) - Prostate Cancer
63
OF
83
DE
85
OF
87
OF
89
TS - Radiation Therapy
93
OF
91
OF
95
OF
97
Illness Occurrence
Broken Ankle
99
OF
101
DE
103
OF
105
OF
107
Illness Occurrence
Broken Ankle
111
OF
109
DE
113
OF
115
2 Years

FIG._4

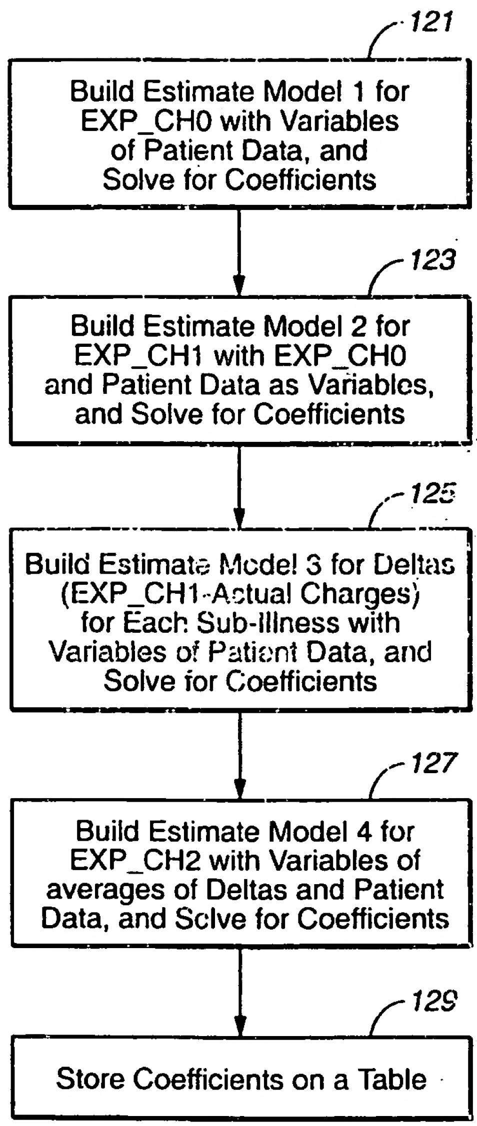
121
Build Estimate Model 1 for EXP_CH0 with Variables of Patient Data, and Solve for Coefficients
123
Build Estimate Model 2 for EXP_CH1 with EXP_CH0 and Patient Data as Variables, and Solve for Coefficients
125
Build Estimate Model 3 for Deltas (EXP_CH1-Actual Charges) for Each Sub-Illness with Variables of Patient Data, and Solve for Coefficients
127
Build Estimate Model 4 for EXP_CH2 with Variables of averages of Deltas and Patient Data, and Solve for Coefficients
129
Store Coefficients on a Table

FIG._6

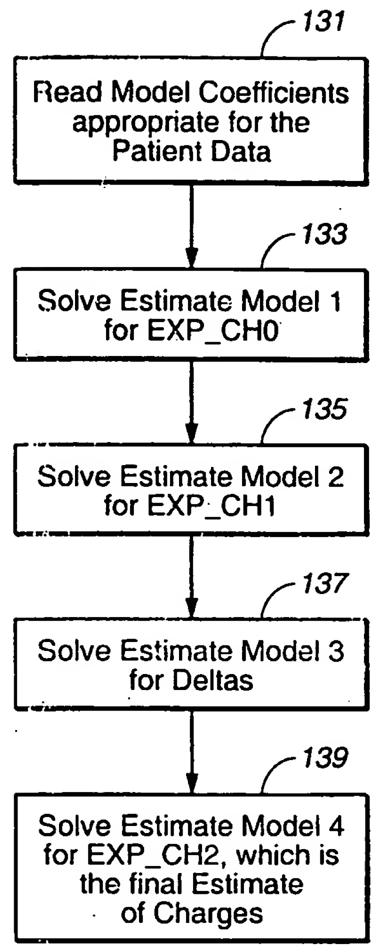
131
Read Model Coefficients appropriate for the Patient Data
133
Solve Estimate Model 1 for EXP_CH0
135
Solve Estimate Model 2 for EXP_CH1
137
Solve Estimate Model 3 for Deltas
139
Solve Estimate Model 4 for EXP_CH2, which is the final Estimate of Charges

FIG._7

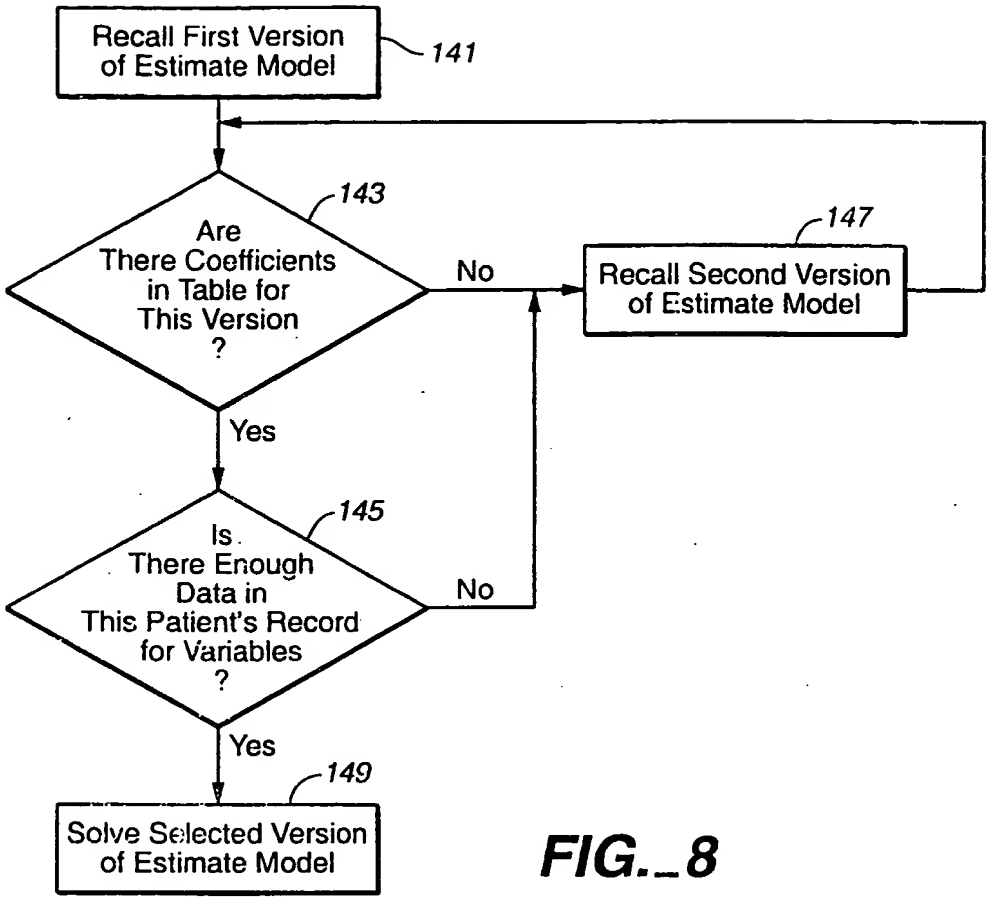
Recall First Version of Estimate Model
141
Are There Coefficients in Table for This Version ?
143
No
Recall Second Version of Estimate Model
147
Yes
Is There Enough Data in This Patient's Record for Variables ?
145
No
Yes
149
Solve Selected Version of Estimate Model

FIG._8

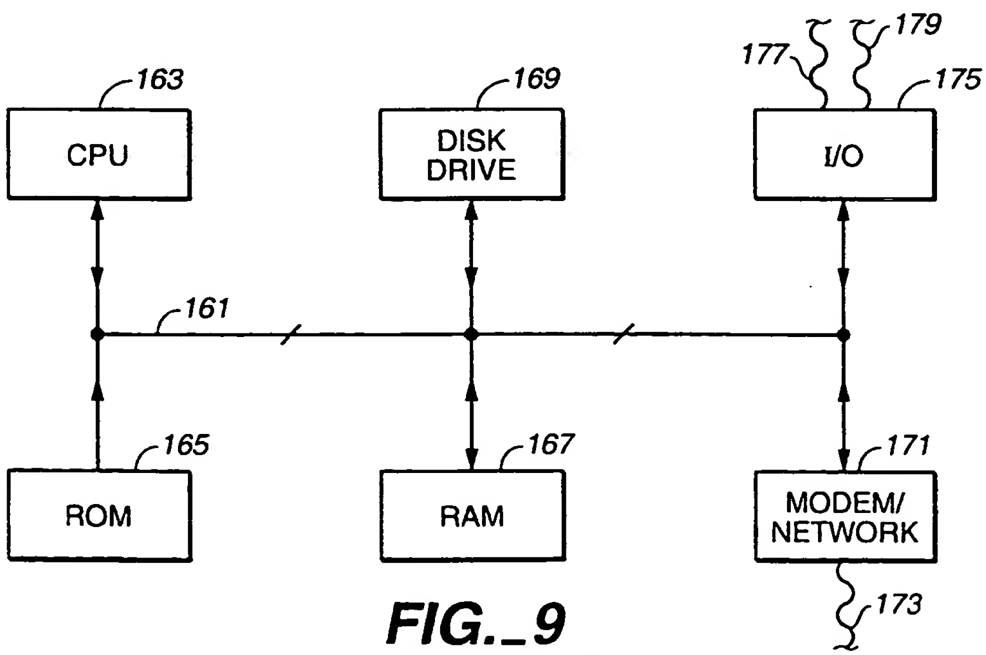
163
CPU
169
DISK DRIVE
177
179
175
I/O
161
165
ROM
167
RAM
171
MODEM/ NETWORK
173

FIG._9

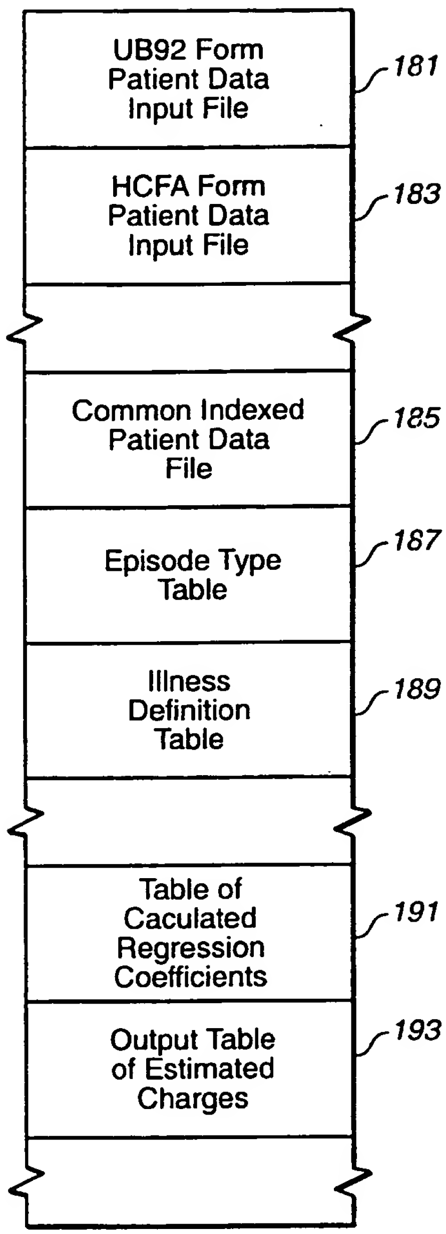
UB92 Form Patient Data Input File
181
HCFA Form Patient Data Input File
183
Common Indexed Patient Data File
185
Episode Type Table
187
Illness Definition Table
189
Table of Caculated Regression Coefficients
191
Output Table of Estimated Charges
193

FIG._10

APPENDIX

In re Patent Application of

QUINN WHITING-O'KEEFE

SERIAL NO.: UNASSIGNED

FILED: HEREWITH

TITLE: TECHNIQUES FOR ESTIMATING CHARGES OF DELIVERING HEALTHCARE SERVICES THAT TAKE COMPLICATING FACTORS INTO ACCOUNT

HOPS Algorithm

901 Mariners Island Boulevard •Suite 565 • San Mateo CA • 94404

1. INTRODUCTION

This document describes the HOPS algorithms.

2. EPISODE CREATION

2.1. Step 1

Note that the calculations are presented as if they are performed once on a large file. For practical reasons, the input file is broken into sections and the calculations performed on each of the sections individually. Finally, just before the actual estimations the individual files are merged together into the appropriate large input file to the estimator. Where appropriate the partial files are referred to as a filename with an appended &i, where the &i is assumed to vary over a set uniquely identifying each of the partial files. Breaking up the input file like this does not affect the calculations and is done for performance reasons. Where the &i is used as part of the filename the reader can essentially ignore it.

This step extends the input file by adding the skey field if it is not already present. The skey field is 0 for the first record and increases by 1 for each subsequent record.

The following fields are dropped from the input file:

- ext_id
- t

The following field is added to the input file:

- skey identifies the service

2.2. Step 2

Create a dataset, &p.ex, comprising an expanded input file with one record for each drg, dx and pr code in the input file, &infn. &p.ex comprises the following fields:

- **skey**
- **cd**
- **cd_doma**
- **cd_src**
- drg
- st_date
- end_date
- billtype
- charges

- person_k

The values of cd_doma, cd_src and cd are defined in the following table:

cd_doma	cd_src	cd
"drg"	"drg"	drg
txdom	"pr0" … "pr7"	pr
"icd9Dx"	"dx0" … "dx9"	dx

2.3. Step 3

Create a table &p.t1 by left joining &p.ex to ref.ClEpiTy on cd_doma (domainid) and cd (itemid). &p.t1 comprises the following fields:

- skey
- cd
- cd_doma
- cd_src
- drg
- st_date
- end_date
- billtype
- charges
- person_k
- epi_ty
- epiTyPri
- epiTy
- epiTyDur

2.4. Step 4 Assign epi_tyx

Create a dataset, &p.t2, by copying &p.t1 to create a single record for each skey. epi_tyx is assigned by processing the &p.t1 records by grouping them by skey and epiTyPri and implementing the following algorithm:

For each record with a given skey repeat steps 1, 2 and 3 until epi_tyx has been assigned.

1. if drg is non-null then set epi_tyx to "IP"
2. if epiTy <> "" and charges >= epiTyDur and epiTy = "IP" then set epiTy and epi_tyx to "DE". This is a temporary fixup until the class definitions are assigned properly.
3. if epiTy <> "" and charges >= epiTyDur and epiTy <> "IP" or "DE" then set epi_tyx to epiTy
4. If epi_tyx remains unassigned after the last record for the skey group then set epi_tyx to "OF".

There is one &p.t2 record for each skey. &p.t2 comprises the following fields:

- **skey**
- epi_tyx

Note that Quinn was attempting to reimplement this step but the work was unfinished. It needs to be completed. The current version of the code uses the original version.

Note that the use of epiTyDur as a charge is extremely funky and needs to be revisited as part of the correction.

2.5. Step 5 Append epiTyPri

Create a table, tmp9, by selecting all distinct records grouped by epiTy from ref.ClEpiTy. tmp9 comprises the following fields:

- **epiTy**
- **epiTyPri**

tmp9 comprises the following data:

epiTy	epiTyPri
DE	mc
IP	ma
OF	md
TS	mb

Create a table, &p.t3, by joining &p.t2 to tmp9 on epi_tyx (epiTy). &p.t3 comprises the following fields:

- **skey**
- epi_tyx

- epiTyPri

Create the dataset, &p.t4, by merging the &infn and &p.t3 files by skey keeping every record in &infn. The following assignments are made:

- epi_ty = epi_tyx
- neg_end= - end_date

&p.t4 comprises the following fields:

- input file variables
- skey
- epi_ty
- epiTyPri
- neg_end negative of the end date

2.6. Step 6 Assign epi_key

Create a dataset, &p.t5, by copying the dataset &p.t4 (grouped by person_k, st_date, neg_end, old_pri = epiTyPri) and assigning each record to an episode as follows:

- for each patient and encounter start date identify the record with the longest encounter duration and assign a new episode key to that record.
- All other records for the same patient and encounter start date are then assigned the episode key, epi_ty and epiTyPri as the longest duration record with the highest priority.

&p.t5 comprises the following fields:

- input file variables
- **epi_key identifies the episode**
- **skey identifies the service**
- epi_ty
- epiTyPri

2.7. Step 7 Link adjacent IP records

Create a dataset, &p.t6, by copying the dataset &p.t5 grouped by person_k and st_date attributing an inpatient episode to an earlier inpatient episode if:

- it starts within 20 days of the start and 3 days of the end of the earlier episode
- it starts before the end of the earlier episode

Also non-inpatient episodes that start before the end of an inpatient episode are assigned to that episode. Note that this assignment may be incomplete in terms of variables such as epiTyPri, etc.

&p.t6 comprises the following fields:

- input file variables
- epi_key identifies the episode
- skey identifies the service
- epi_ty
- epiTyPri

2.8. Step 8

Create a dataset, &p.x, comprising an expanded &p.t6 with one record for each drg, dx and pr code in &p.t6. &p.x comprises the following fields:

- skey
- cd
- cd_doma
- cd_src
- epi_ty
- epi_key

The values of cd_doma, cd_src and cd are defined in the following table:

cd_doma	cd_src	cd
"drg"	"drg"	drg
txdom	"pr0" … "pr7"	pr
"icd9Dx"	"dx0" … "dx9"	dx

2.9. Step 9 Join with ClEpiSubTy

Create a table &p.1 by left joining &p.x to ref.clepisub on epi_ty, cd_doma (domainid) and cd (itemid). &p.1 comprises the following fields:

- skey
- cd
- cd_doma

- cd_src
- epi_ty
- epi_key
- epiSubTy episode subtype
- epiSubPr episode subtype priority
- epiSubDu episode subtype duration

Note that there may be multiple rows per skey since there may be multiple values of epiSubTy for each unique choice of epi_ty, domainid and itemid in ref.clepisub. However the values of epiSubPri and epiSubDur are the same for each value of epiSubTy.

Note that in practice there are very few entries for 'OF' epi_ty in the clepisub table.

2.10. Step 10

Create a table, &p.2, comprising one record for each service by selecting the first record for each skey, grouped by skey and epiSubPr, that has a non-null epiSubTy. If no record satisfies the criteria then the last record for the skey is output with the following values:

- epiSubTy = '?'
- epiSubPr = 'z'
- epiSubDu = 150

&p.2 comprises the following fields:

- skey identifies the service
- epiSubTy episode subtype
- epiSubPr episode subtype priority
- epiSubDu episode subtype duration
- epi_key identifies the episode

2.11. Step 11

Create a dataset, &p.3, by copying &p.2 and forcing each record for a given epi_key to have the same values for epiSubPr, epiSubTy and epiSubDu as the record for that epi_key with the highest priority epiSubPr. &p.3 comprises the following fields:

- skey identifies the service
- epiSubTy episode subtype

- epiSubPr episode subtype priority
- epiSubDu episode subtype duration
- epi_key identifies the episode

2.12. Step 12

Create a table, &p.t7, by inner joining &p.t6 to &p.3 on skey. &p.t7 comprises the following fields:

- input file variables
- **epi_key** **identifies the episode**
- **skey** **identifies the service**
- epi_ty
- epiTyPri
- epiSubTy episode subtype
- epiSubPr episode subtype priority
- epiSubDu episode subtype duration

2.13. Step 13

Create the dataset, &p.en1, by copying the dataset &p.t7 grcuped by epi_ty, person_k, epiSubTy and st_date. Each episode of type "TS" that occurs within epiSubDu days of an earlier "TS" episode is subsumed into that earlier episode by assigning the epi_key of the earlier record to the later record.

&p.en1 comprises the following fields:

- input file variables
- **epi_key** **identifies the episode**
- **skey** **identifies the service**
- epi_ty
- epiTyPri
- epiSubTy episode subtype
- epiSubPr episode subtype priority
- anal_sub episode subtype

3. ILLNESS ASSIGNMENT

3.1. Step 1

Create a dataset, &p.ex, comprising an expanded &p.en1 with one record for each drg, dx and pr code in &p.en1. &p.ex comprises the following fields:

- skey
- cd
- cd_doma
- cd_src
- drg
- st_date
- end_date
- billtype
- charges
- person_k
- epi_ty

The values of cd_doma, cd_src and cd are defined in the following table:

cd_doma	cd_src	cd
"drg"	"drg"	drg
txdom	"pr0" … "pr7"	pr
"icd9Dx"	"dx0" … "dx9"	dx

3.2. Step 2

Create a table, &p.t1, by inner joining &p.ex to ref.BaseSev on cd_doma (domainId) and cd (itemId). &p.t1 comprises the following fields:

- skey
- cd
- cd_doma
- cd_src

- drg
- st_date
- end_date
- billtype
- charges
- person_k
- epi_ty
- illprior
- illId
- assocDur
- assocLev
- factorTy

3.3. Step 3

Create a dataset, &p.t2, by copying &p.t1 grouped by person_k, illId, assocLev and st_date. The purpose of this dataset is to remove those records with assocLevel = "1" that do not have an associated assocLevel = "0" illness within +/- assocDur days. Additionally, a complex indexing field is calculated and added to &p.t2.

The copying logic is as follows:

- copy all records with factorTy = "S"
- copy all records with assocLevel = "0"
- copy all records with assocLevel = "1" for which an assocLevel "0" illness with the same illId has started within assocDur days before or after st_date.
- copy all records with assocLevel = "2"

The indexing logic is as follows:

assocLevel	Conditions	ordVar
0	epi_ty = 'IP' + cd_src = 'drg'	factorTy + 'X' + illPrior + 'b' + '0dr'
0	epi_ty = 'IP' + cd_src = 'dx0'	factorTy + 'X' + illPrior + 'b' + '0dx'
0	otherwise	factorTy + 'Y' + illPrior + 'b' + tmp
1	epi_ty = 'IP' + cd_src = 'drg'	factorTy + 'X' + illPrior + 'a' + '0dr'

1	epi_ty = 'IP' + cd_src = 'dx0'	factorTy + 'X' + illPrior + 'a' + '0dx'
1	otherwise	factorTy + 'Y' + illPrior + 'a' + tmp
2	-	factorTy + 'Z' + illPrior + 'c' + tmp

where

- illPrior means substr(illPrior, 1, 2)
- tmp is npr when cd_src is prn (i.e. pr5 -> 5pr)
- tmp is ndx when cd_src is dxn (i.e. dx5 -> 5dx)

&p.t2 comprises the following fields:

- skey
- cd
- cd_doma
- cd_src
- drg
- st_date
- end_date
- billtype
- charges
- person_k
- epi_ty
- illprior
- illId
- assocDur
- assocLev
- factorTy
- ordVar

3.4. Step 4

Create a dataset, &p.t3, by copying &p.t2 grouped by person_k, st_date, skey, and ordvar. The purpose of this is to bundle the records back into skey level records and assign the il0-il9 and iltype0-iltype9 variables for the record. Note that the illId field is assigned to il0-9 and factorTy is assigned to iltype0-9.

The il0-il9 and associated type are assigned as follows for a given skey:

1. process the records in ordvar order
2. assign the first record found with assocLev 0 or 1 to il0 and iltype0. Note that the 1's are processed before the 0's because of the ordering of the data in ordVar.
3. assign subsequent records with assoclev of 0 to subsequent ilx values where x is 1...9, and similarly assign iltypex.
4. for assocLevel 2 records where there is no il0 record and a preceding record (diagnosis, procedure, drg) occurred within assocDur days of the current record then the preceding record is assigned to il0.

Note that assocLev 1 and 2 records are only ever assigned to il0 and never to any higher ilx. Furthermore assocLev 2 records are only attributed to il0 if another record occurred with assocDur days of the current record.

&p.t3 comprises the following fields:

- skey
- il0
- ... iix, where x = 1...8
- il9
- iltype0
- ... iltypex, where x = 1...8
- iltype9
- il0prior priority of the il0 record

3.5. Step 5

Create a table, &p.t4, by left joining &p.cn1 to &p.t3 on skey. &p.t4 comprises the following fields:

- input file variables
- epi_key identifies the episode
- skey identifies the service
- epi_ty
- epiTyPri
- epiSubTy episode subtype
- epiSubPr episode subtype priority
- il0prior priority of the il0 record

- il0
- ... ilx, where x = 1...8
- il9
- iltype0
- ... iltypex, where x = 1...8
- iltype9

3.6. Step 6

Create a table, &p.t6, by copying &p.t4 grouped by epi_key. &p.t6 contains one record per episode and comprises the following fields:

- epi_key identifies the episode
- epi_ill the episode illness
- epi_end end date of the episode

Each record with a given epi_key is processed and the charges for each il0 are summed. The il0 with the largest total charges over all records for the episode is assigned to epi_ill.

Note that epi_end is set to the latest end_date of all services in the episode.

3.7. Step 7

Create a table, &p.t7, by left joining &p.t6 to ref.ilOcDur on epi_ill (illId). &p.t7 comprises the following fields:

- epi_key identifies the episode
- epi_ill the episode illness
- epi_end end date of the episode
- chrnAct chronic or acute flag
- ilOcDur

Note that the chrnAct flag takes the following values:

C chronic

A acute

0 one of the following five illnesses:

- Reason for Consult
- Psychiatric Exam
- General Medical Exam

- Vaccine
- Prophylactic Measures

3.8. Step 8

Create a table, &p.en2, by inner joining &p.t4 to &p.t7 on epi_key and where &p.t4.il0 <> “” and &p.t7.epi_ill <> “”. &p.en2 comprises the following fields:

- input file variables
- epi_key identifies the episode
- skey identifies the service
- epi_ty
- epiTyPri
- epiSubTy episode subtype
- epiSubPr episode subtype priority
- il0prior priority of the il0 record
- il0
- ... ilx, where x = 1...8
- il9
- iltype0
- ... iltypex, where x = 1...8
- iltype9
- epi_ill
- chrnAct chronic or acute flag
- ilOcDur
- epi_end

3.9. Step 9

Create the dataset, &p.clas_&i, by copying the dataset &p.en2 grouped by person_k, epi_ill and st_date. This step identifies each illness occurrence for a patient. The copying progresses by processing each record for a given person_k and epi_ill in st_date order and assigning each record with the same illness to the same illness occurrence provided

- st_date is within ilOcDur days of the illness occurrence start date

  and

- st_date is before the end date of the latest episode in the illness occurrence.

Otherwise a new illness occurrence is initiated for that illness.

&p.clas_&i comprises the following fields:

- input file variables
- epi_key identifies the episode
- skey identifies the service
- epi_ty
- epiTyPri
- epiSubTy episode subtype
- epiSubPr episode subtype priority
- il0prior priority of the il0 record
- il0
- ... ilx, where x = 1...8
- il9
- iltype0
- ... iltypex, where x = 1...8
- iltype9
- epi_ill
- chrnAct chronic or acute flag
- ilOcDur
- ilOcKey identifies the illness occurrence
- ill_st unused

4. CREATION OF INPUT FILES FOR ESTIMATOR

4.1. Step 1

Create a dataset, &p.z1, by copying the dataset &p.clas_&i grouped by epi_key. &p.z1 contains one record for each episode and comprises the following fields:

- age — patient's age
- charges — total charge for the episode
- drg — drg for the episode
- doer_md — doer_md with the largest charge
- pcp_md — pcp_md with the largest charge
- dx0-dx9 — 10 diagnoses with the largest charge
- end_date — latest end_date of any service in the episode
- end_elig
- epiSubPr — epiSubPri with the largest charge
- epiSubTy — epiSubTy with the largest charge
- epiTyPri
- epi_ty
- il0-il9 — 10 illnesses with the largest charges
- ord_md — ord_md with the largest charge
- person_k — patient identifier
- pr0-pr7 — 8 procedures with the largest charge
- sex — sex of the patient
- st_date — earliest start date of any service in the episode
- st_elig
- end_elig
- skey — unique identifier for episode
- txdo0-txdo7 — domain identifier for pr0-pr7
- iltype0-iltype9 — iltype for 10 illnesses with largest charge
- ilOcKey — uniquely identifies the illness occurrence
- ilOcDur
- chrnAct — specifies whether the episode is chronic or acute

The elements defined as being those with the *largest charge* are all calculated by summing up the charges for these items across all services in an episode and choosing the n items with the largest charge. The dx_i, pr_i and il_i fields are stored in order of decreasing charges.

4.2. Step 2

Create a dataset, &p.z2, by copying &p.z1 grouped by person_k and st_date. &p.z2 comprises one record for each episode.

This step adds the collateral illness list to the illness list for each episode. The step is implemented by tracking two moving windows of time: a 20 day window and a 200 day window.

Illnesses that occur sufficiently often in each window are added to the il0-9 and iltype0-9 fields for the episode. Sufficiently often means 2 or more times for the 20 day window and 5 or more times for the 100 day window. A primary illness for the episode (il0) counts as 2 times so that such illnesses are always added.

&p.z2 comprises the following fields:

- age — patient's age
- charges — total charge for the episode
- drg — drg for the episode
- doer_md — doer_md with the largest charge
- pcp_md — pcp_md with the largest charge
- dx0-dx9 — 10 diagnoses with the largest charge
- end_date — latest end_date of any service in the episode
- end_elig
- epiSubPr — epiSubPri with the largest charge
- epiSubTy — epiSubTy with the largest charge
- epiTyPri
- epi_ty
- il0-il9 — 10 illnesses with the largest charges along with important illnesses within a surrounding 20 and 100 day window
- ord_md — ord_md with the largest charge
- person_k — patient identifier
- pr0-pr7 — 8 procedures with the largest charge
- sex — sex of the patient
- st_date — earliest start date of any service in the episode

- st_elig
- end_elig
- skey — unique identifier for episode
- txdo0-txdo7 — domain identifier for pr0-pr7
- iltype0-iltype9 — iltype for 10 illnesses with largest charge
- ilOcKey — uniquely identifies the illness occurrence
- ilOcDur
- chrnAct — specifies whether the episode is chronic or acute

4.3. Step 3

Create a table, tmp9, by selecting fields from &p.z2 grouped by person_k and il0. tmp9 comprises the following fields:

- **person_k** — patient identifier
- **il0** — illness identifier
- dur_ill — illness duration in days

dur_ill is equal to *max(end_elig) - min(st-date)* for all the episodes for the person. This table contains the durations for each (primary) illness occurrence for each person.

Create a table, tmp9, from tmp9 by updating the dur_ill field as follows:

1. set dur_ill to 40 if dur_ill < 40
2. set dur_ill to dur_ill / 365.25

tmp9 comprises the following fields:

- **person_k** — patient identifier
- **il0** — illness identifier
- dur_ill — illness duration in years

Note that illness duration is truly a duration for an illness and is NOT a duration for the illness occurrence. There is a single value for each illness.

Create a table, &p.epi_0, by inner joining tmp9 to &p.z2 on person_k and il0 keeping all fields in &p.z2 and the dur_ill field from tmp9. &p.epi_0 comprises the following fields:

- age — patient's age
- charges — total charge for the episode
- drg — drg for the episode
- doer_md — doer_md with the largest charge

- pcp_md pcp_md with the largest charge
- dx0-dx9 10 diagnoses with the largest charge
- end_date latest end_date of any service in the episode
- end_elig
- epiSubPr epiSubPri with the largest charge
- epiSubTy epiSubTy with the largest charge
- epiTyPri
- epi_ty
- il0-il9 10 illnesses with the largest charges along with important illnesses within a surrounding 20 and 290 day window
- ord_md ord_md with the
- person_k patient identifier
- pr0-pr7 8 procedures with the largest charge
- sex sex of the patient
- st_date earliest start date of any service in the episode
- st_elig
- end_elig
- skcy unique identifier for episode
- txdo0-txdo7 domain identifier for pr0-pr7
- iltype0-iltype9 iltype for 10 illnesses with largest charge
- ilOcKey uniquely identifies the illness occurrence
- ilOcDur
- chrnAct specifies whether the episode is chronic or acute
- dur_ill illness duration in years

4.4. Step 4

The purpose of this step is to create the files used in the estimation. The step is performed eight times, parameterized by the values of modTy and epiTy provided in the following table:

modTy	epiTy
IO	OF
IO	DE

IO	OD
IO	AL
IL	OF
IL	DE
IL	OD
IL	AL

The codes have the following meaning:

- **IO** illness occurrence
- **IL** patient/illness
- **OF** office
- **DE** day encounter
- **OD** office and day encounter
- **TS** therapeutic series
- **IP** inpatient
- **AL** office + day encounter + therapeutic series + inpatient

IO and IL correspond to the two ways of looking at the data and is primarily driven by the need to differentiate between acute care and chronic care. The main difference is follow-up treatment is physician discretionary in acute care but is not discretionary for chronic care. The illness-occurrence view of the data is used for acute illnesses of a finite duration and the patient-illness view is used for chronic illnesses.

The eight output files are as follows with the name derived from the parameterizing modTy and epiTy:

- &oLib..IOOF_&i
- &oLib..IODE_&i
- &oLib..IOOD_&i
- &oLib..IOAL_&i
- &oLib..ILOF_&i
- &oLib..ILDE_&i
- &oLib..ILOD_&i
- &oLib..ILAL_&i

Note that the &i indicate that the files are subsets of a larger file.

The decomposition into the eight files is to allow the following estimates to be performed in a later stage:

By episode (calculate $ per episode)

- inpatient (IP)
- therapeutic series (radiotherapy) (TS)
- therapeutic series (chemotherapy) (TS)
- day encounter (DE)

By illness occurrence (calculate $ per illness occurrence))

- office (OD)
- office + day encounter (OD)
- office + day encounter + therapeutic series + inpatient (AL)

By illness (calculate $ per year)

- office (OF)
- office + day encounter (OD)
- office + day encounter + therapeutic series + inpatient (AL)

4.4.1. Stage 1

Initialize the variables byStr and byVar depending on the value of modTy as follows:

modTy	byStr	byVal
IO	ilOcKey	ilOcKey
IL	person_k il0	il0

4.4.2. Stage 2

Initialize the variable exec1 depending on the value of epiTy as follows:

epiTy	exec1
OF	%str(epi_ty = "OF")
DE	%str(epi_ty = "DE")
DE or OF (i.e. OD)	%str(epi_ty = "OF" or epi_ty = "DE")
AL	1

For each of the eight cases, create the appropriately named output file by copying the file epi_&i grouped by the value of byStr. Note that only those records satisfying the exec1 rule contribute to the calculations and are copied to the output file. For example, for ILOF estimates the output file comprises one record for each person_k, il0 group that had at least one non-zero record of 'OF' episode type.

The output files comprise the following fields:

- person_k patient identifier
- sex sex of the patient
- skey unique identifier for episode??
- age patient's age
- st_date earliest st_date for the group
- end_date latest end_date of any service in the group
- st_elig
- end_elig
- charges total charge for the group
- drg drg with the highest weighted charge for the group
- ord_md ord_md with the highest charge for the group
- doer_md doer_md with the highest charge for the group
- pcp_md pcp_md with the highest charge for the group
- dx0-dx9 diagnoses with the highest weighted charges for the group
- epi_ty
- il0-il9 illnesses with the highest weighted charges for the group
- iltype0-9 illness types for the illnesses in il0-9
- pr0-pr7 procedures with the highest weighted charges for the group
- txd0-7 domain type for the procedures in pr0-7
- dur_ill
- il0cDur
- num_il0c
- chrnAct specifies whether the episode is chronic or acute

Note that the ord_md, doer_md and pcp_md assignment uses actual charges not weighted charges. **Furthermore the charge weightings for the il0-9, dx0-9 and pr0-7 assignments are all different.**

If one of doer_md or ord_md is missing then its value is set to the value of the other.

Note that there appears to be some magic in the calculations of the following variables:

- il0-9
- iltype0-9
- dx0-9

This magic specifically concerns seemingly arbitrary numbers and diagnoses being used in the calculations.

4.5. Step 5

Create the final output data files by merging the intermediate working tables as follows:

- &oLib..IOOF_&i -> &oLib..IOOF
- &oLib..IODE_&i &oLib..IODE
- &oLib..IOOD_&i &oLib..IOOD
- &oLib..IOAL_&i &oLib..IOAL
- &oLib..ILOF_&i &oLib..ILOF
- &oLib..ILDE_&i &oLib..ILDE
- &oLib..ILOD_&i &oLib..ILOD
- &oLib..ILAL_&i &oLib..ILAL

These are the input files to the estimation process.

5. ESTIMATION

The following models are estimated in the estimation phase:

- IOOF illness occurrence, office
- IOOD illness occurrence, office + day encounter
- IOAL illness occurrence, all
- ILOF illness, office
- ILOD illness, office + day
- ILAL illness, all
- EPDE episode, day encounter
- EPTS episode, therapeutic series
- EPIP episode, inpatient

The estimation step comprises eight stages named as follows:

- A0
- A1
- A2
- B1
- B2
- C1
- C2
- C3

The estimation phase is controlled by a set of primary and secondary controlling variables which are set appropriately for each stage to parameterize the operation of a single set of functions that are invoked for all the estimates.

The primary controlling parameters are:

- **mv** model variables (regressors)
- **bv** grouping or join variables

The secondary controlling parameters are:

- **pr** regression rows are kept if the probability of non-zero R2 is <= pr_*. No rows are excluded if pr_* = 1.
- **rq** Regression rows are kept if adjrsq > rq_*. No rows are excluded If rq_* = -2.

- **rc** regression rows are kept if <number of parameters> * rc_* <= EDF. No regression is done if rc_* = -2.
- **av** average rows are kept if stderr = "." or abs(avg) - stderr * av_* >= 0. No rows are excluded if av_* = 0.
- **cn** average rows are kept if count >= cn_*. No rows are excluded if cn_* = 0. No average is calculated if cn_* = -2.
- **dn** controls whether or not a stage is processed. Dn_* is zero if either:
  rc_* = -2 and cn_* = -2
  or
  rq_* = 1 and av_* = 0

Each controlling parameter has a value for each stage so that the complete set of controlling variables is the Cartesian product of the controlling parameters and the stages. The following are example of controlling variables and their values:

- mv_A0 numdx
- mv_C3 numdx adj_sum adj_avg adj_neg adj_min
- bv_A0 il0 dx0
- bv_C3 col_dx

The estimation itself is finally controlled by the following variables:

- infns specifies the input file
- outfn specifies the output file
- modelLib specifies the model library
- bld_mod specifies whether a model should be built
- modTy specifies the type of model: episode (ep), illness (il), illness occurrence (io).
- epi_ty episode type: office (of), day encounter (de), therapeutic series (TS), inpatient (IP)
- pre5 filename prefix
- debug specifies the level of debug output

Processing is primarily controlled by the dn_A0 … dn_C3 variables which are used to specify whether or not a given stage of processing is performed. Additionally, the bld_mod variable is used to identify those pieces of the algorithm which are only executed when a new model is being built.

The following steps are performed to calculate the various estimates.

5.1. Initialize the Controlling Variables

Initialize the controlling variables to the values described in the appropriate table in the appendix. The following different tables of controlling variables are defined in the appendix:

- EPIP
- EPDE & EPTS
- IL
- IO
- NotIL

5.2. Generate Formula for Calculating Estimates

For each of the model variables, mv_A0 ... mv_C4 generate the variables sz_A0 ... sz_C4 and fo_A0 ... fo_C4. The sz variables contain the function that sets each model variable field to zero if it is missing. The fo variables contain the function that will calculate the estimate for the model (by taking the sum of products of the model parameters and the actual data).

5.3. Create the Dataset &p.t0

Create the dataset, &p.t0, by copying the input file, &infns and adding some fields. &p.t0 comprises the following fields:

- &infns fields
- mx_il_in — number of illnesses minus one
- mx_il_sq — square of mx_il_in
- dur_sq — square of dur_ill
- dur_mxil — product of dur_ill and mx_il_in
- numdx — number of diagnoses minus one
- numdx_sq — square of numdx
- dur_dx — product of numdx and dur_ill
- mx_dx — product of numdx and mx_il_in
- sex1 — gender as coded values {1, 2}

Additionally the fields st_elig and end_elig are validated so that they fall within reasonable bounds.

The records copied to &p.t0 depend on the type of data being processed as defined by the modTy as described in the following.

For modTy = "IL" only those records with chrnact of "C" are copied and the following fields are added to the dataset:

- dur_ill average of end_elig - st_date and end_date - st_date
- zerolos set to 1 if los > 0 otherwise set to 0

For modTy = "EP" only those records with epi_ty = "&epi_ty" are copied

For modTy = "IO" only those records with chrnact other than "C" are copied and the following fields are added to the dataset:

- ilOcLen end_date minus st_date
- mx_len product of mx_il_in and ilOcLen
- ilOcl_sq square of ilOcLen
- dx_len product of num_dx and ilOcLen
- dx_il_len product of dxil_num and ilOcLen.

The variable &ainfn is set to &p.t0.

5.4. A0, A1 and A2 Processing (Model only)

This section describes the processing of the A0, A1 and A2 stages executed only when building a new model.

If dn_A0 is non-zero invoke the AppFiles macro with the following parameters:

- depVar charges
- infn &ainfn
- stage A0
- outfn &tabPre.A0
- debug &debug

This calculates charges as a function of the model variables (mv_A0) using the group variables (bv_A0) on the dataset &ainfn.

If dn_A1 is non-zero invoke the AppFiles macro with the following parameters:

- depVar charges
- infn &ainfn
- stage A1
- outfn &tabPre.A1
- debug &debug

This calculates charges as a function of the model variables (mv_A1) using the group variables (&bv_A1) on the dataset &ainfn.

If dn_A2 is non-zero invoke the AppFiles macro with the following parameters:

- depVar charges
- infn &ainfn
- stage A2
- outfn &tabPre.A2
- debug &debug

This calculates charges as a function of the model variables (mv_A2) using the group variables (&bv_A2) on the dataset &ainfn.

5.5. A0, A1 and A2 Processing

This section describes the processing of the A0, A1 and A2 stages that are always executed.

The variable &tmpNm is set to the value &ainfn

If dn_A0 is non-zero create the table, &p.A0, by left joining &tmpNm to &tabPre.A0 on the variables in bv_A0. The variable &tmpNm is set to the value &p.A0. &p.A0 comprises the following fields:

- &tmpNm fields
- exp_A0 estimate of charges from A0 model

If dn_A1 is non-zero create the table, &p.A1, by left joining &tmpNm to &tabPre.A1 on the variables in &bv_A1. The variable &tmpNm is set to the value &p.A1. &p.A1 comprises the following fields:

- &tmpNm fields
- exp_A1 estimate of charges from A1 model

If dn_A2 is non-zero create the table, &p.A2, by left joining &tmpNm to &tabPre.A2 on the variables in &bv_A2. The variable &tmpNm is set to the value &p.A2. &p.A2 comprises the following fields:

- &tmpNm fields
- exp_A2 estimate of charges from A2 model

Create the dataset, &p.A3, by copying &tmpNm and setting exp_ch0, the first estimate of the charges, to the appropriate choice of exp_A0, exp_A1 or exp_A2 as follows:

- exp_A0 if exp_A0 >= 10
- exp_A1 otherwise and exp_A1 >= 10

- exp_A2 otherwise and exp_A2 >= 0

The record is discarded if there are no exp_A0, exp_A1 and exp_A2 estimates.

&p.A3 comprises (essentially) the following fields:

- &ainfn fields
- exp_ch0 first estimate of charges

The variable &binfn is set to the value &p.A3 if any of dn_A0, dn_A1 or dn_A2 are non-zero and is otherwise set to &ainfn.

5.6. B1 and B2 Processing (Model Only)

This section describes the processing of the B1 and B2 stages executed only when building a new model.

If dn_B1 is non-zero invoke the AppFiles macro with the following parameters:

- depVar charges
- infn &binfn
- stage B1
- outfn &tabPre.B1
- debug &debug

This calculates charges as a function of the model variables (mv_B1) using the group variables (&bv_B1) on the dataset &binfn.

If dn_B2 is non-zero invoke the AppFiles macro with the following parameters:

- depVar charges
- infn &binfn
- stage B2
- outfn &tabPre.B2
- debug &debug

This calculates charges as a function of the model variables (mv_B2) using the group variables (&bv_B2) on the dataset &binfn.

5.7. B1 and B2 Processing

This section describes the processing of the B1 and B2 stages that are always executed.

The variable &tmpNm is set to the value &binfn

If dn_B1 is non-zero create the table, &p.B1, by left joining &tmpNm to &tabPre.B1 on the variables in &bv_B1. The variable &tmpNm is set to the value &p.B1. &p.B1 comprises the following fields:

- & ainfn fields
- exp_ch0 first estimate of charges
- exp_B1 estimate of charges from B1 model

If dn_B2 is non-zero create the table, &p.B2, by left joining &tmpNm to &tabPre.B2 on the variables in &bv_B2. The variable &tmpNm is set to the value &p.B2. &p.B2 comprises the following fields:

- & ainfn fields
- exp_ch0 first estimate of charges
- exp_B2 estimate of charges from B2 model

Create a dataset, &p.B3, by copying &tmpNm and setting exp_ch1, the second estimate of the charges, to the appropriate choice of exp_B1, exp_B2 or exp_ch0 as follows:

- exp_B1 if 10 < exp_B1 < (20 * exp_ch0)
- exp_B2 otherwise and 10 < exp_B2 < (20 * exp_ch0)
- exp_ch0 **if dn_A1 or dn_A2 are non-zero. !?**

The record is discarded if all three of these tests fail.

&p.B3 comprises (essentially) the following fields:

- &ainfn fields
- exp_ch0 first estimate of charges
- exp_ch1 second estimate of charges

The variable &cinfn is set to the value &p.B3 if any of dn_B1 or dn_B2 are non-zero and is otherwise set to &binfn.

5.8. C1, C2, C3 and C4 Processing (Collateral Adjustments)

This section describes the processing of the C1, C2, C3 and C4 stages. This section is performed only if dn_C1 or dn_C2 are non-zero. Some parts are performed depending on the values of dn_C3 and dn_C4.

Create a dataset, &p.ex, by copying the dataset &cinfn and creating one record for each illness or diagnosis in the &cinfn record depending on the value of the associated ilOrDx field. &p.ex comprises the following fields:

- skey identifies the records
- delta1 exp_ch1 - charges

- &bv_C1 grouping variables
- &mv_C1 model variables
- col_dx if present is equal to dx[i]
- col_ildx if present is equal to either "dx" + dx[i] or "il" + il[i]
- col_il if present is equal to il[i]

Set the variable &tmpfn to &p.ex.

If bld_mod and dn_C1 are non-zero invoke the AppFiles macro with the following parameters:

- depVar delta1 (= exp_ch1 - charges)
- infn &p.ex
- stage C1
- outfn &tabPre.C1
- debug &debug

This calculates delta1 as a function of the model variables (mv_C1) using the group variables (bv_C1) on the dataset &p.ex.

If bld_mod and dn_C2 are non-zero invoke the AppFiles macro with the following parameters:

- depVar delta1
- infn &p.ex
- stage C2
- outfn &tabPre.C2
- debug &debug

This calculates delta1 as a function of the model variables (mv_C2) using the group variables (bv_C2) on the dataset &p.ex.

If dn_C1 is non-zero create the table, &p.t4, by left joining &p.ex to &tabPre.C1 on the variables in &bv_C1. The variable &tmpNm is set to the value &p.t4. &p.t4 comprises the following fields:

- &p.ex fields
- delta_2 estimate of charges from C1 model

If dn_C2 is non-zero create the table, &p.t5, by left joining &tmpNm to &tabPre.C2 on the variables in &bv_C2. The variable &tmpNm is set to the value &p.t5. &p.t5 comprises the following fields:

- &p.ex fields

- delta_3 estimate of charges from C2 model

Create the dataset, &p.t6, by copying &tmpNm grouped by skey. &p.t6 comprises the following fields:

- &p.ex fields
- adj_sum sum of deltas for the group
- adj_neg sum of negative deltas for the group
- adj_avg average delta for the group
- adj_max maximum delta for the group
- adj_min minimum delta for the group

where for each record in the group the delta is chosen as follows:

- delta_2 if present
- delta_3 otherwise, if present
- 0 otherwise

Create the table, &p.t7, by left joining &cinfn to &p.t6 on skey and keeping the following fields:

- &ainfn fields
- exp_ch0 first estimate of charges
- exp_ch1 second estimate of charges
- adj_sum sum of deltas for the group
- adj_neg sum of negative deltas for the group
- adj_avg average delta for the group
- adj_max maximum delta for the group
- adj_min minimum delta for the group

Create the dataset, &p.t8, by copying &p.t7 setting missing values of the adj_sum, adj_neg, adj_avg, adj_max and adj_min fields to zero. &p.t8 comprises the following fields:

- &ainfn fields
- exp_ch0 first estimate of charges
- exp_ch1 second estimate of charges
- adj_sum sum of deltas for the group
- adj_neg sum of negative deltas for the group
- adj_avg average delta for the group
- adj_max maximum delta for the group

- adj_min minimum delta for the group
- delta1 exp_ch1 - charges

If bld_mod and dn_C3 are non-zero invoke the AppFiles macro with the following parameters:

- depVar delta1
- infn &p.t8
- stage C3
- outfn &tabPre.C3
- debug &debug

This calculates delta1 as a function of the model variables (mv_C3) using the group variables (bv_C3) on the dataset &p.t8.

If bld_mod and dn_C4 are non-zero invoke the AppFiles macro with the following parameters:

- depVar delta1
- infn &p.t8
- stage C4
- outfn &tabPre.C4
- debug &debug

This calculates delta1 as a function of the model variables (mv_C4) using the group variables (bv_C4) on the dataset &p.t8.

If dn_C3 is non-zero create the table, &p.t9, by left joining &p.t8 to &tabPre.C3 on the variables in &bv_C3. The variable &tmpNm is set to the value &p.t9. &p.t9 comprises the following fields:

- &cinfn fields
- adj_sum sum of deltas for the group
- adj_neg sum of negative deltas for the group
- adj_avg average delta for the group
- adj_max maximum delta for the group
- adj_min minimum delta for the group
- delta1 exp_ch1 - charges
- del_C3 estimate of delta1 from C3 model

If dn_C4 is non-zero create the table, &p.t10, by left joining &p.t9 to &tabPre.C4 on the variables in &bv_C4. The variable &tmpNm is set to the value &p.t10. &p.t10 comprises the following fields:

- &ainfn fields
- exp_ch0 first estimate of charges
- exp_ch1 second estimate of charges
- adj_sum sum of deltas for the group
- adj_neg sum of negative deltas for the group
- adj_avg average delta for the group
- adj_max maximum delta for the group
- adj_min minimum delta for the group
- delta1 exp_ch1 - charges
- del_C3 estimate of delta1 from C3 model
- del_C4 estimate of delta1 from C4 model

Create the dataset, &p.t11, by copying &tmpNm and setting exp_ch2, the third estimate of the charges, to the appropriate value chosen as follows:

- exp_ch1 - del_C3
  if exp_ch1 - del_C3 > 10
- exp_ch1 - del_C4
  otherwise, if exp_ch1 - del_C4 > 10
- exp_ch1
  otherwise

&p.t11 comprises the following fields:

- &ainfn fields
- exp_ch0 first estimate of charges
- exp_ch1 second estimate of charges
- exp_ch2 final estimate of the charges
- delta1 exp_ch1 - charges
- del_C3 estimate of delta1 from C3 model
- del_C4 estimate of delta1 from C4 model

The variable &dinfn is set to the value &p.t11 if any of dn_B1 or dn_B2 are non-zero and is otherwise set to &cinfn.

5.9. Add Display Variables

Create a dataset, &p.t12, by copying & dinfn

This step adds variables used in calculating percentiles and partial charges and some variables used for display purposes.

&p. t12comprises the following fields:

- age
- charges
- chrnact
- doer_md
- drg
- dur_ill
- dx0 ... dx9
- end_date
- end_elig
- episubty
- epi_ty
- il0 ... il9
- ilocdur
- iltype0 ... iltype9
- maxil
- ord_md
- pcp_md
- person_k
- pr0 ... pr7
- sex
- skey
- st_date
- st_elig
- txdo0 ... txdo7
- exp_ch0 first estimate of charges
- exp_ch1 second estimate of charges
- exp_ch2 final estimate of the charges
- delta1 exp_ch1 - charges
- del_C3 estimate of delta1 from C3 model
- del_C4 estimate of delta1 from C4 model

- exp_chg   exp_ch2 * adjustment factor (= 1)
- delta   exp_chg - charges
- exp_csq   exp_chg * exp_chg
- fps   financial performance score
  100 * delta / max(exp_chg, charges)
- xdel   delta / dur_ill if modty = "IL", otherwise delta
- xexp   exp_chg / dur_ill if modty = "IL", otherwise
  exp_chg
- xchg   charges / dur_ill if modty = "IL", otherwise charges
- prim_cla   drg - if modty = "EP" and epity = "IP"
  epiSubTy - if modty = "EP" and epity = "DE"
  epiSubTy + "|" + il0 - otherwise if modty = "EP"
  il0, otherwise

5.10. Percentiles

The purpose of this step is to create an percentile ordered list of xexp.

Create the table, &p.t1, from &p.t12 grouped by the variables in prim_cla. &p.t1 comprises the following fields:

- &p.t12 fields
- totcnt   total number of records for each prim_cla group

Create the table, &p.t13, from &p.t1. &p.t13 contains an ordered list of xexp values and percentiles for each prim_cla group. A new entry is inserted in &p.t13 each time a new value or new percentile value is encountered. The table therefore allows any percentile value to be identified for the expected charge (xexp). &p.t13 comprises the following fields:

- prim_cla fields   fields contained in the prim_cla
- pctlVal   xexp value for the percentile bucket
- pctlPct   percentile bucket
- accp   percentile bucket

&p.t13 is stored as the ILOF_AP, IOOP_AP or EPIP_AP table. Note that the percentile charges are not re-estimated.

5.11. Partial Charge Estimates (Procedure Class Estimates)

The purpose of this step is to perform partial-charge estimates which means estimates of the total charges for the following categories: medical, surgical, radiology, laboratory and referral. These will be referred to as procedure class

charges. The assignment to the various category is made by reference to a table providing a one-to-one mapping between CPT4 and ICD9 procedure codes and a category. This step is performed only if dn_P1 is 1 and &partials is 1.

Create the table, Clas, by copying the file Clas_0. In practice, since the 0 indicates only the first batch of potentially up to 30 batches this step combines each batch file, Clas_i, into a single file Clas.

Create the table, &p.1, by left joining &p.t13 to Clas on person_k and il0 (epi_ill) and the following condition:

&p.t13.st_date <= Clas.st_date <= &p.t13.end_date

Note that this condition may be wrong since it does not include Clas.end_date. Further it appears to be needed because identifying fields have been dropped from working tables allowing the records to be identified simply and accurately between the two files.

&p.1 comprises the following fields:

- skey
- charges
- exp_chg
- exp_csq
- prim_cla
- epi_ty
- zzchg Clas.charges
- zztxdom Clas.txdom
- zzpr0 ... zzpr7 Clas.pr0 ... Clas.pr7
- bpcp_md Clas.pcp_md
- bord_md Clas.ord_md
- bepi_ty Clas.epi_ty

Create a dataset, &p.2, by copying &p.1 keeping only those records for which epi_ty = "OD" and bepi_ty = "OF" or "DE". The practical effect of this is to consider primarily those records containing CPT4 procedure codes. &p.2 is created by inserting a new record for each procedure in the &p.1 table and adding two variables: vProcChg and prx. VProcChg is set to zzchg divided by the number of procedures in the record. Note that this is potentially very funky since, for example, lab charges are allocated the same amount as surgical procedures which is probably not valid.

&p.2 comprises the following fields:

- skey

- charges
- exp_chg
- exp_csq
- prim_cla
- epi_ty
- zztxdom Clas.txdom
- bpcp_md Clas.pcp_md
- bord_md Clas.ord_md
- bepi_ty Clas.epi_ty
- prx zzpr[i]
- vProcChg zzchg / number of procedures

Create a table, &p.3, by left joining &p.2 to ClProc on prx (itemId) and zztxdom (domainid). ClProc is the refernece table that associates the procedure classes with the CPT4 and ICD9 procedure codes. &p.3 comprises the following fields:

- skey
- charges
- exp_chg
- exp_csq
- prim_cla
- epi_ty
- zztxdom Clas.txdom
- bpcp_md Clas.pcp_md
- bord_md Clas.ord_md
- bepi_ty Clas.epi_ty
- prx zzpr[i]
- vProcChg zzchg / number of procedures
- vProcClas partial charge class for prx

Create a dataset, &p.t14, by copying &p.t3 grouped by skey and vProcCla. For each skey one record is inserted into &p.t14 comprising the total referral charges found by summing up those records for that skey for which bpcp_md <> bord_md. Additionally, one record is inserted in &p.t14 for each procedure class with non-zero total charges for that service. Note that the sum of the procedure class charges must be equal to the total charge for the service but that the referral charges are separate and a single record could contribute to a procedure charge

and a referral charge. This is not regarded as double counting since it reflects the reality that a referral physician could perform the procedure and consequently incur a charge. &p.14 comprises the following fields:

- skey
- charges
- exp_chg
- exp_csq
- prim_cla
- epi_ty
- bepi_ty Clas.epi_ty
- vProcClas procedure class
- vProcChg total charge for the procedure class

If bld_mod is non-zero invoke the AppFiles macro with the following parameters:

- depVar vProcChg
- infn &p.t14
- stage P1
- outfn &p.t15
- debug &debug

This calculates vProcChg as a function of the model variables (mv_P1) using the group variables (bv_P1) on the dataset &p.t14.

Copy &p.t15 to the output dataset &tabPre.P1.

Create a table, &p.t15, by left joining &p.t14 to &tabPre.P1 on the variables in &bv_P1. &p.t15 comprises the following fields:

- skey
- charges
- exp_chg
- exp_csq
- prim_cla
- epi_ty
- bepi_ty Clas.epi_ty
- vProcClas procedure class
- vProcChg total charge for the procedure class
- vProcExp estimated charge for the procedure class

Note that for each value of skey there may be several rows, one for each value of vProcClas.

Create a dataset, &p.t16, by copying the dataset &p.t15 grouped by skey

- skey
- vlab actual laboratory charge
- wlab estimated laboratory charges
- vmed actual medical charges
- wmed estimated medical charges
- voth actual other charge for
- woth estimated other charges
- vrad actual radiology charges
- wrad estimated radiology charges
- vref actual referral charges
- wref estimated referral charges
- vsur actual surgery charges
- wsur estimated surgery charges
- vprocsum total non-referral charges

Note that the &p.t16 dataset is essentially a transposed version of &p.t15 where the procedure charges are stored by field rather than by row. The estimated charges are set to 10 if the estimate was less than 10.

Create the table, &outfn, by left joining &p.t13 to &p.t16 on skey. &outfn is the final output file from the estimation process and the actual name is of the form E_ILOF, etc.

6. APPENDIX

6.1. Controlling Variables - Primary

The following distinct sets of controlling variables are defined:

- EPIP
- EPTS & EPDE
- IL
- IO
- NotIL

The following variables are used in the definitions of the controlling variables:

- numdx — number of diagnoses
- exp_ch0 — estimate of expected charge from A phase
- numdx_sq — numdx * numdx
- mx_il_in — number of illnesses minus one
- mx_il_sq — mx_il_in * mx_il_in
- mx_dx — numdx * mx_il_in
- adj_sum — sum of deltas from collateral illnesses calculation
- adj_avg — average delta from collateral illnesses calculation
- adj_neg — sum of negative deltas from collateral illnesses calculation
- adj_min — minimum delta from collateral illnesses calculation
- exp_ch1 — estimate of expected charge from B phase
- delta — exp_ch1 - charges
- dxil_num — numdx + mx_il_in + 1
- dxil_sq — dxil_num * dxil_num
- dxil_len — dxil_num * ilOcLen
- ilOcLen — end_date - st_date
- ilOcL_sq — ilOcLen * ilOcLen

6.1.1. EPIP

Stage	Est	Ave	Dependent Variable	Group	Model
A0	n	y	charges	dx0, pr0	numdx
A1	n	y	charges	drg, pr0	numdx
A2	n	y	charges	drg	numdx
B1	y	n	charges	drg	exp_ch0, age, numdx, numdx_sq, mx_il_in, mx_il_sq, mx_dx
C1	n	y	delta	drg, col_dx	numdx
C2	n	y	delta	col_dx	numdx
C3	y	n	charges	drg	numdx, adj_sum, adj_avg, adj_neg, adj_min

6.1.2. EPTS & EPDE

Stage	Est	Ave	Dependent Variable	Group	Model
A0	n	y	charges	pr0, il0	numdx
A1	n	y	charges	pr0, epiSubTy	numdx
A2	n	y	charges	epiSubTy	numdx
B1	y	n	charges	epiSubTy, il0	exp_ch0, age, numdx, numdx_sq, mx_il_in, mx_il_sq, mx_dx
B2	y	n	charges	epiSubTy	exp_ch0, age, numdx, numdx_sq, mx_il_in, mx_il_sq, mx_dx
C1	y	y	delta	col_ildx, epiSubTy	numdx, exp_ch1
C2	y	y	delta	col_ildx	numdx, exp_ch1
C3	y	n	charges	epiSubTy	numdx, adj_sum, adj_avg, adj_neg, adj_min

6.1.3. IL

Stage	Est	Ave	Dependent Variable	Group	Model
A0	y	n	charges	dx0, il0	age, numdx, numdx_sq, dur_ill, dur_sq, dur_dx
A1	y	n	charges	il0	age, numdx, numdx_sq, dur_ill, dur_sq, dur_dx
A2	y	y	charges	il0	numdx, dur_ill
B1	y	n	charges	dx0	exp_ch0, numdx, dur_ill
C1	y	y	delta	col_dx, il0	exp_ch1, numdx
C2	y	y	delta	col_dx	exp_ch1, numdx
C3	y	n	charges	col_dx	numdx, adj_sum, adj_avg, adj_neg, adj_min

6.1.4. IO

Stage	Est	Ave	Dependent Variable	Group	Model
A0	n	y	charges	dx0	dxil_num
A1	n	y	charges	il0	dxil_num
B1	y	n	charges	il0	age, exp_ch0, dxil_num, dxil_sq, ilocLen, ilocL_sq, dxil_len
B2	y	n	charges	il0	dxil_num
C1	n	y	delta	col_ildx, il0	dxil_num
C2	n	y	delta	col_ildx	dxil_num
C3	y	n	charges	il0	dxil_num, adj_sum, adj_avg, adj_neg, adj_min

6.1.5. NotIL

Stage	Est	Ave	Dependent Variable	Group	Model
A0	y	n	charges	dx0, il0	age, dxil_num, dxil_sq, dur_ill, dur_sq, dur_dxil

A1	y	n	charges	il0	age, dxil_num, dxil_sq, dur_ill, dur_sq, dur_dxil
A2	y	y	charges	il0	dxil_num, dur_ill
B1	y	n	charges	dx0	exp_ch0, dxil_num, dur_ill
C1	n	y	delta	col_ildx, il0	numdx
C2	n	y	delta	col_ildx	numdx
C3	y	n	charges	il0	dxil_num, adj_sum, adj_avg, adj_neg, adj_min

6.2. Controlling Variables - Secondary

6.2.1. EPIP

	A0	A1	A2	B1	B2	C1	C2	C3	C4
ay	3	3	2	0	0	5	5	0	0
bv	dx0 pr0	pr0 drg	drg	drg	il0	*	col_dx	drg	dx0
cn	5	5	5	-2	-2	20	20	-2	-2
dn									
mv	numdx	numdx	numdx	*	*	numdx	numdx	*	byvar
pr	.01	.01	.01	.01	.01	.01	.01	.01	.001
rc	-2	-2	-2	8	-2	-2	-2	-8	-2
rq	1	1	1	.05	1	1	1	.01	1

- bv_C1 col_dx drg
- mv_B1 mx_dx exp_ch0 age numdx numdx_sq mx_il_in mx_ilsq
- mv_B2 exp_ch0 numdx
- mv_C3 EPDE numdx adj_sum adj_avg adj_neg adj_min EPTS &

6.2.2. IL

	A0	A1	A2	B1	B2	C1	C2	C3	C4

av	0	0	2	0	0	3	3	0	0
bv	dx0 il1	il0	il0	dx0	il0	col_dx il0	col_dx	il0	dx0
cn	-2	-2	5	-2	-2	10	10	-2	-2
dn									
mv	*	&mv_A0	*	*	none	*	*	*	&mv_C3
pr	.01	.01	.01	.01	.01	.001	.001	.01	.001
rc	20	8	8	20	-2	15	15	8	-2
rq	.1	.01	.01	.3	1	.2	.2	.02	1

- mv_A0 age numdx numdx_sq dur_ill dur_sq dur_dx
- mv_A2 numdx dur_ill
- mv_B1 numdx dur_ill exp_ch0
- mv_C1 numdx exp_ch1
- mv_C2 numdx exp_ch1
- mv_C3 numdx adj_sum adj_avg adj_neg adj_min

6.2.3. IO

	A0	A1	A2	B1	B2	C1	C2	C3	C4
av	2	2	0	0	0	3	3	0	0
bv	dx0	il0	il0	il0	il0	*	col_ildx	il0	dx0
cn	5	5	-2	-2	-2	10	10	-2	-2
dn									
mv	dxil_num	dxil_num	dxil_num	*	dxil_num	dxil_num	dxil_num	*	&mv_C3
pr	.01	.01	.01	.01	.01	.007	.01	.05	.001
rc	-2	-2	-2	8	8	-2	-2	8	-2
rq	1	1	1	.01	.05	1	1	.01	1

- bv_C1 col_ildx il0

- mv_B1 age exp_ch0 dxil_num dxil_sq ilOcLen ilOcL_sq dxil_len
- mv_C3 dxil_num adj_sum adj_avg adj_neg adj_min

6.2.4. NotIL

	A0	A1	A2	B1	B2	C1	C2	C3	C4
av	0	0	2	0	0	3	3	0	0
by	dx0 il1	il0	il0	dx0	il0	col_ildx il0	col_ildx	il0	dx0
cn	-2	-2	5	-2	-2	10	10	-2	-2
dn									
mv	*	&mv_A0	*	*	none	numdx	numdx	*	&mv_C3
pr	.01	.01	.01	.01	.01	..007	.01	.05	.001
rc	20	8	8	20	-2	15	15	8	-2
rq	.1	.01	.01	.3	1	1	1	.01	1

- mv_A0 age dxil_num dxil_sq dur_ill dur_sq dur_dxil
- mv_A2 dxil_num dur_ill
- mv_B1 dxil_num dur_ill exp_ch0
- mv_C3 dxil_num adj_sum adj_avg adj_neg adj_min

6.3. AppFiles Macro

The purpose of the AppFiles macro is to use linear regression to generate an estimate of a dependent variable from a given input file and put the regression coefficients in an output file. A stage variable acts an index into several external tables of parameters which specify the variables and parameters used in the analysis.

In essence the AppFiles macro is a special-purpose, parameterizable, linear regression calculator that ensures that exactly the same logic is used on the various regressions involved in performing the HOPS estimates.

The AppFiles macro has the following inputs:

- depVar - the dependent variable to be calculated

- infn - names the input file
- stage - identifies a set of external variables
- outfn -names the output file
- debug -specifies if debug output is to be generated

The AppFiles macro comprises the following steps:

6.3.1. Initialize fnA and fnB

Create empty datasets, fnA and fnB, comprising the following fields:

- bv_&stage - variables specified in the external table &bv_&stage where stage is the input variable.
- _adjrsq_
- _edf_
- _rsq_
- avgCnt
- intercep

6.3.2. Perform First Regression

Invoke proc reg to calculate &depvar as a function of the variables specified in &mv_&stage. The input file is &infn and the output file is fnA.

6.3.3. Calculate Average

Append to the table, fnB, by selecting records from &infn grouped by the variables specified in &bv_&stage. The following fields are calculated:

- &bv_&stage
- intercep average of &depvar
- avgCnt number of records in the group
- stderr standard error of &depVar

6.3.4. Merge Regression and Average

Create a dataset, tmpApp2, by copying fnA and fnB keeping only those records that satisfy certain criteria. During the copy process a field inXXX is added that is set to 0 for fnA records and 1 for fnB records.

6.3.5. Sort tmpApp2 dataset

Sort the dataset, tmpApp2, by &bv_&stage and inXXX. This means that fnA records for a given &bv_&stage occur before those of fnB.

6.3.6. Select one record for each set of &bv_&stage variables

Create a dataset, &outfn, by copying tmpApp2 keeping the first record found for each group of records with equal &bv_&stage variables. This means that the regression set fnA is used in preference to fnB.

6.4. Reference Tables

6.4.1. ClEpiTy

field name	comment
epiTy	(DE, IP, OF, TS)
domainId	(UB92RevCd, cpt4, icd9Dx)
itemId	
epiTyPri	(ma, mb, mc, md)
epiTyDur	(0, 30, 100, 175, 250, 300, 1000)

6.4.2. ClEpiSub

field name	comment
epiTy	(DE, OF, TS)
epiSubTy	410 values
domainId	(UB92RevCd, cpt4, icd9Dx, icd9Pr)
itemId	
epiSubPri	(ma, mm, mn, n, nn, pa)
epiSubDur	(0, 100, 300)

Note that the epiSubTy field should not be in the primary key since the values of the dependent fields epiSubPri and epiSubDur do not depend on the value of

epiSubTy independently of the values of epiTy, domainId and itemId. The epiTy, epiSubTy relationshop should be in a different table.

Note that there can be many entries with the same epiTy and epiSubTy distinguished by the value of domainId and itemId.

What is the true distinction between ClEpiTy and ClEpiSub. The Pri and Dur field values differ for equivalent itemId and domainId.

6.4.3. BaseSev

illnessPriority assocLevel assocDur priority

field name	comment
illnessId	995 values
domainId	(UB92RevCd, cpt4, drg, icd9Dx, icd9Pr)
itemId	15000 values
illnessParent	36 values
factorTy	(I, S) [S - only for assocLevel -0]
illnessPriority	(mf, mm, nm, tb, td, u)
assocLevel	(0, 1, 2) 2 only for illnessId = 'StdLab'
assocDur	(0, 10, 30, 50, 100)
priority	(m, mm, tb, td)